Naval Research Laboratory
Washington, DC 20375-5320

NRL/MR/6350--99-8402

Interpolation-Based Condensation Model Reduction Part 1: Frequency Window Reduction Method Application to Structural Acoustics

R. P. INGEL
L. D. FLIPPEN, JR.

Multifunctional Materials Branch
Materials Science and Technology Division

C. T. DYKA

GEO-Centers, Inc.
Fort Washington, MD

September 30, 1999

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave Blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 30, 1999	

4. TITLE AND SUBTITLE

Interpolation-Based Condensation Model Reduction
Part 1: Frequency Window Reduction Method Application to Structural Acoustics

5. FUNDING NUMBERS

6. AUTHOR(S)

R.P. Ingel, L.D. Flippen, Jr., and C.T. Dyka

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/MR/6350--99-8402

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
Washington, DC 20375-5320

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

*GEO-Centers, Inc., Fort Washington, MD

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

A

13. ABSTRACT (*Maximum 200 words*)

Degree of freedom model reduction and frequency windowing via interpolation are combined for application to acoustic finite element analysis. Projection operators are employed for the model reduction or condensation process. Interpolation is then introduced over a user defined frequency window, which can have real and imaginary boundaries and be quite large. Hermitian (which require derivative information) interpolation functions as well as standard Lagrangian functions, which can be linear, quadratic or cubic, have been used to construct the interpolation windows. For 1D windows, the location of interior evaluation points can be varied to increase accuracy. Excellent results are obtained for the example problems, even for large degree of freedom (DOF) reduction and large frequency windows containing numerous resonances (eigenvalues). The solution for the reduced models within the interpolation windows accurately indicate the location of the resonances. The program FWR (Frequency Window Reduction) is written in Fortran 90 and has the ability to read in third party generated finite element models.

14. SUBJECT TERMS

Degree of freedom reduction
FEM model reduction
Acoustics
Frequency sweeps

15. NUMBER OF PAGES

162

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std 239-18
298-102

Table of Contents

INTERPOLATION-BASED CONDENSATION MODEL REDUCTION PART 1: FREQUENCY WINDOW REDUCTION METHOD APPLICATION TO STRUCTURAL ACOUSTICS

1 Introduction

The efficient and accurate solution of large degree of freedom (DOF) problems remains difficult for linear, dynamic and steady state analyses. Many problems have become too large and complex to solve in an economic fashion, especially if, for example, a large number of frequency responses are required for a structural acoustic analysis. In addition, the problem of the interaction of scales, which can extend from the micro to the macro-scale, has not been adequately addressed. This is especially true for heterogeneous media such as composite materials and structures. For linear problems, the use of sub-structuring methods [1,2] can be very effective. However, when applied to acoustic steady-state analyses, sub-structuring methods introduce further approximations and still require the solution of the reduced problem at each desired frequency.

Overall, finite element method (FEM) programs have increased in capabilities and in usage with improvements in computational platforms. The application of FEM analyses to large complex problems has significantly risen as the need for more detailed and sophisticated analyses and designs has increased. Computationally based analysis is utilized in many fields where full experimental analysis is difficult to obtain. Thus computer modeling is relied on as there is no other method of obtaining and testing designs. The costs of both experimental and computational analyses are a major driving factor for improved techniques for large complex analyses. FEM computational tools which could reduce the computational costs of very large complex models and which provide enhanced analysis capabilities are desperately needed.

Flippen [3,4] has developed a general-purpose dynamic condensation model reduction (CMR) theory which utilizes projection operators on the governing differential equations. This approach allows general DOF reduction in spatially discretized models of heterogeneous material and is not restricted to periodic media or other homogenization assumption. The CMR method is applicable to general second order differential equations in time. This methodology [5] was implemented into a linear finite element environment and applied to dynamic analysis (time domain) of transient elasticity problems.

Manuscript approved September 10, 1999.

In reference [6], Flippen extended this condensation model reduction (CMR) theory to include frequency domain analysis. The use of projection operators on the governing differential equations allowed interpolation across selected frequency windows to be accomplished. Hermitian interpolation functions [1], which required derivative information on the boundaries of the frequency window, were used. The CMR method was applied to structural acoustics analysis and was implemented into a one dimensional (1D) finite volume environment. The method was applied to a complex 1D composite structure and good results were obtained across wide frequency windows for the reduced DOF solutions as compared to the full DOF solution. The eigenvalues of the reduced system were found to match quite well with those of the full system, even for significant DOF reduction. In this particular paper, a method was developed (though not implemented) to account for singularities caused by resonances using singularity decomposition techniques. In a recent effort, Flippen[7] has extended this work into a general model reduction theory, which unites and generalizes the reduced basis and frequency window class of methods as submethods.

In this present work, the CMR method has been extended to frequency windowing directly into a 2D and 3D finite element method (FEM) environment and applied to structural acoustics. The approach is referred to as the Frequency Window Reduction (FWR) method for degree of freedom reduction of spatially discrete time-transformed linear models. In this case frequency windows are chosen in which a discretized model can be reduced in its degrees of freedom. The method consists of discretizing a FEM model and selecting the "master" node points (DOFs) which are to be retained in the analysis. The contribution of the "slave" node points, which are to be condensed out, is reformulated and is used to generate an interpolation of the functional dependence of the "slave" response across a selected frequency window. A "reduced" system operator now may be constructed, which consists of the "master" DOFs, and the influence of the interpolated "slave" response. The reduced system response can then solved across a selected frequency interpolation window.

In addition to the Hermitian interpolation functions (two-point interpolation) of the "slave" response first employed with this method, standard Lagrangian interpolation functions (linear, quadratic and cubic not requiring derivative calculations) have been included. These lat-

ter functions, which do not require derivative information at the evaluation points, are an important addition that are far simpler to implement, very efficient and provide function evaluation within the window of interest that can increase accuracy. Also for the standard quadratic and cubic interpolation schemes, the location of the evaluation points that are within the interior of the frequency window can be varied for a one dimensional (1D) frequency window (a line in frequency space). This is an important capability that can substantially increase the accuracy within selected regions within the interpolation window. The use of evaluation points within the window was extended to include higher order Hermitian functions with one and two interior interpolation evaluation points for 1D windows. This has led to a further improvement in accuracy.

Computational analyses were performed using a program developed in Fortran 90 which can read in and solve third party generated finite element models. One- and two-dimensional results from the FWR program are presented and show excellent results for the "reduced" system solutions as compare to the full DOF solutions. Accurate results are obtained for large DOF reduction and large frequency windows even with the presence of numerous resonances (eigenvalues) within the window, which have not been explicitly accounted for in the analysis as yet.

2 Frequency Window Reduction Method (FWR)

The generalize equation of ODEs can be written in compact form as

$$\mathbf{L}\,u = f \tag{2.1}$$

where **L** is an n by n matrix of operators, u is an n by 1 column matrix (vector) of the system response and f is an n by 1 column matrix (vector) of the system stimuli. The value of n represents the total number of degrees of freedom (DOF) for the system. In order to examine the frequency domain the exponential form of the Fourier transform, $F = e^{i\omega t}$, is applied to equation (2.1). The operator $\hat{\mathbf{L}}$ is now a function of the continuous independent variable ω, which generally is complex. Note that $i = \sqrt{-1}$. Equation (2.1) can then be rewritten as

$$\hat{\mathbf{L}}\,\hat{u} = \hat{f} \tag{2.2}$$

and the response and stimuli terms are

$$\hat{u} = F(u) \tag{2.3a}$$

$$\hat{f} = F(f) \tag{2.3b}$$

Applying P , a permutation matrix operator, on the operator matrix $\mathbf{L}$ in equation (2.1) or $\hat{\mathbf{L}}$ in equation (2.2) as $\tilde{\mathbf{L}} = P\mathbf{L}P^{-1}$ such that $\mathbf{L}$ (or $\hat{\mathbf{L}}$) can now be partitioned as

$$\tilde{\mathbf{L}} = \begin{pmatrix} \tilde{\mathbf{L}}_{11} & \tilde{\mathbf{L}}_{12} \\ \tilde{\mathbf{L}}_{21} & \tilde{\mathbf{L}}_{22} \end{pmatrix} \tag{2.4}$$

where the index "1" represents the "master" or independent degrees of freedom and the index "2" represents the "slave" or dependent degrees of freedom.

The term "master" degree of freedom refers to the independent DOFs of the reduced system in the FEM model. The "masters" DOFs are typically chosen as DOFs, which have applied loads or constraints such as fixed boundary conditions in an FEM model. In addition to loaded and constrained DOFs, the master DOFs may also be chosen as important nodes represented in the models; for example, the interfaces or boundaries between element groups of differing material properties or element grouping midpoints. They may also be selected as DOFs where the resultant system response is of interest to examine.

The term "slave" degree of freedom refers to the dependent DOFs or those DOFs, which are to be condensed or reduced out of the full FEM model. These are DOFs, which typically do not have applied loads or constraints. They are also DOFs which do not contribute significantly to the overall system response or are not of major interest.

Further applying the permutation matrix to the system stimuli according to

$$\mathbf{P} = P^{-1}\begin{pmatrix} I & 0 \\ \alpha & 0 \end{pmatrix}P \tag{2.5}$$

results in the system stimuli being partitioned as

$$\tilde{f} = \mathbf{P}\, f = \begin{pmatrix} \tilde{f}_m \\ \tilde{f}_s \equiv \alpha\, \tilde{f}_m \end{pmatrix} \tag{2.6}$$

Here the subscripts "m" and "s" refer to the "master" and "slave" DOFs respectively. The matrix α is defined such that $\tilde{f}_s = \alpha\, \tilde{f}_m$ and I is the Identity Matrix.

The α matrix, the "slave-master" response matrix, is an operator matrix, which is used to relate "slave" DOF properties such as constraints, forces and applied loads to the "master" DOFs. Usually "slaves" DOFs are chosen such that they do *not* have constraints or loads. However, for very large complex FEM models in order to reduce the total number of DOFs , it may desirable to also condense out DOFs which have constraints or applied loads. Hence the α operator matrix can be used to interpolate from the "master" DOFs for example an applied load or boundary condition to the "slave" DOFs. In this present work it has been assumed that the α operator matrix is zero based on the choice of the "master" and "slave" DOFs, However a non-zero α operator matrix as well as completely arbitrary constraint and loading conditions of the DOFs will be the subject of future work.

The condensation model reduction (CMR) theory has shown that the general DOF reduction of an operator matrix is

$$\tilde{L}_{21} + \tilde{L}_{22}\,\beta = \alpha\left(\tilde{L}_{11} + \tilde{L}_{12}\,\beta\right) \tag{2.7}$$

where $\tilde{L} = \begin{pmatrix} \tilde{L}_{11} & \tilde{L}_{12} \\ \tilde{L}_{21} & \tilde{L}_{22} \end{pmatrix}$ is the reordered /partitioned operator matrix and the α matrix, the "slave-master" response matrix is defined. The goal is to solve for the β matrix in order to construct the reduced operator.

Rearranging equation (2.7)

$$\left(\tilde{L}_{22} - \alpha\tilde{L}_{12}\right)\beta = \alpha\tilde{L}_{11} - \tilde{L}_{21} \tag{2.8}$$

The β matrix can be found directly as

$$\beta \equiv G^{-1} H \tag{2.9}$$

where the G^{-1} matrix is the matrix inverse

$$G \equiv \tilde{\mathbf{L}}_{22} - \alpha \tilde{\mathbf{L}}_{12} \tag{2.10}$$

and the H matrix is

$$H \equiv \alpha \tilde{\mathbf{L}}_{11} - \tilde{\mathbf{L}}_{21} \tag{2.11}$$

Having determined the form of β, the final "*reduced*" operator, $\overline{\mathbf{L}}_0$, is

$$\overline{\mathbf{L}}_0 = \tilde{\mathbf{L}}_{11} + \tilde{\mathbf{L}}_{12} \beta \tag{2.12}$$

and hence the reduced system equation to be solved is

$$\overline{\mathbf{L}}_0 \ \mathrm{u}_m = \prod \tilde{\mathrm{f}} \tag{2.13}$$

with $\prod \tilde{\mathrm{f}} = \tilde{\mathrm{f}}_m$ where $\prod = (\mathrm{I} \quad 0)\ \mathbf{P}$.

The column matrix (vector) u_m represents the "master" or independent DOF system response. Finally the "slave" response can be obtained from the "master" response by reconstruction via

$$\mathrm{u}_s = \beta \ \mathrm{u}_m \tag{2.14}$$

The goal in the general CMR method is to reduce the number of degrees of freedom in order to allow efficient computational solutions to a problem but which maintains the accuracy of the full system. The primary consideration is the accuracy of the reduced solution as compared to the full system solution. Hence there will be trade-offs among the degree of reduction, i.e. the number of DOFs condensed out from the full system, the computational efficiency and the accuracy of the reduced solution.

3 FWR Formulation for Mechanical Systems

For the case of a mechanical system equation (2.1) is written as

$$\mathbf{M}\frac{\partial^2 u}{\partial t^2} + \mathbf{D}\frac{\partial u}{\partial t} + \mathbf{S}u = f \tag{3.1}$$

where **M**, **D** and **S** are the mass, damping and stiffness matrices respectively, u is the system displacements and f is the system loads. So the matrix of operators **L** is defined as

$$\mathbf{L} \equiv \mathbf{M}\frac{\partial^2}{\partial t^2} + \mathbf{D}\frac{\partial}{\partial t} + \mathbf{S} \tag{3.2}$$

The frequency domain in obtained from the exponential form of the Fourier transform, $F = e^{i\omega t}$, applied to equation (3.2), so that the operator **L** becomes

$$\hat{\mathbf{L}} \equiv -\omega^2\mathbf{M} + i\omega\mathbf{D} + \mathbf{S} \tag{3.3}$$

The mechanical system operator $\hat{\mathbf{L}}$ is a function of the continuous independent variable ω, and equation (3.1) can then be explicitly rewritten as

$$\hat{\mathbf{L}}(\omega)\,\hat{u}(\omega) = \hat{f}(\omega) \tag{3.4}$$

and the displacement and load terms are $\hat{u} = F(u)$ and $\hat{f} = F(f)$.

The application of the permutation matrix operator to $\hat{\mathbf{L}}$, equation 2.3 as $\tilde{\mathbf{L}} = P\mathbf{L}P^{-1}$ such that it can be partitioned directly, results in the reordering and partitioning of the **M**, **D**, and **S** matrices as

$$\tilde{\mathbf{M}} = \begin{pmatrix} \tilde{\mathbf{M}}_{11} & \tilde{\mathbf{M}}_{12} \\ \tilde{\mathbf{M}}_{21} & \tilde{\mathbf{M}}_{22} \end{pmatrix}, \quad \tilde{\mathbf{D}} = \begin{pmatrix} \tilde{\mathbf{D}}_{11} & \tilde{\mathbf{D}}_{12} \\ \tilde{\mathbf{D}}_{21} & \tilde{\mathbf{D}}_{22} \end{pmatrix} \quad \text{and} \quad \tilde{\mathbf{S}} = \begin{pmatrix} \tilde{\mathbf{S}}_{11} & \tilde{\mathbf{S}}_{12} \\ \tilde{\mathbf{S}}_{21} & \tilde{\mathbf{S}}_{22} \end{pmatrix} \tag{3.5}$$

There are two approaches to obtain the "reordered" global matrices and partitioned matrices based on the number of "master" and "slave" DOFs used in the CMR method. The first is to directly assemble the global matrices from the local matrices in the reordered nodal/DOF numbering. This approach requires an FEM code, which computes and assembles the global mass,

damping and stiffness matrices directly from the element mass, damping and stiffness local matrices. For this, a look-up table of the relationship between the original node numbering and the "master" or reordered node numbering is necessary. Then for a given element and element connectivity in the original numbering, a transformation is used simply to put that local matrix (original numbering) into the global matrix (reordered numbering). The second approach is to reorder according to the "master" node number previously assembled global matrices in the original node numbering via the look-up table. This second approach has the advantage that matrices may be imported from other codes in order to perform the CMR method. (See example Appendix A)

Finally the system operator $\hat{\mathbf{L}}$ matrix equation is obtained from the operator matrix in equation (3.5) in partitioned form as

$$\tilde{\mathbf{L}} \equiv -\omega^2 \begin{pmatrix} \tilde{\mathbf{M}}_{11} & \tilde{\mathbf{M}}_{12} \\ \tilde{\mathbf{M}}_{21} & \tilde{\mathbf{M}}_{22} \end{pmatrix} + i\omega \begin{pmatrix} \tilde{\mathbf{D}}_{11} & \tilde{\mathbf{D}}_{12} \\ \tilde{\mathbf{D}}_{21} & \tilde{\mathbf{D}}_{22} \end{pmatrix} + \begin{pmatrix} \tilde{\mathbf{S}}_{11} & \tilde{\mathbf{S}}_{12} \\ \tilde{\mathbf{S}}_{21} & \tilde{\mathbf{S}}_{22} \end{pmatrix} \tag{3.6}$$

The general CMR method can now be applied to the above equation in order to reduce the number of degrees of freedom.

4 Generalize Formulation of the System Operator, $\mathbf{L}(q)$

Beginning with equation (3.3) the operator matrix, $\hat{\mathbf{L}}$,

$$\hat{\mathbf{L}}(\omega) \equiv -\omega^2 \mathbf{M} + i\omega \mathbf{D} + \mathbf{S} \tag{4.1}$$

can be transformed to a *generalized* complex variable, $q = q_1 + iq_2$ by $q \equiv i\omega$ (Appendix B) so equation (4.1) becomes

$$\hat{\mathbf{L}} \equiv q^2 \mathbf{M} + q\mathbf{D} + \mathbf{S} \tag{4.2}$$

It is assumed that the matrix $\alpha = f(q)$ and that the matrices **M**, **D**, and **S** are <u>not</u> functions of the generalize variable q (or frequency).

Equation (4.2) may then be rewritten as explicitly as a function of the generalize complex variable q in the CMR general DOF reduction form as

$$\left[\tilde{\mathbf{L}}_{22}(q)-\alpha(q)\tilde{\mathbf{L}}_{12}(q)\right]\beta(q)=\ \alpha(q)\tilde{\mathbf{L}}_{11}(q)-\tilde{\mathbf{L}}_{21}(q) \tag{4.3}$$

Expanding in terms of the partitioned mass, damping and stiffness matrices and keeping $\alpha(q)$ as a function of q equation (4.3) becomes

$$\begin{aligned}&\left[\left(q^2\mathbf{M}_{22}+\ q\mathbf{D}_{22}+\ \mathbf{S}_{22}\right)-\alpha(q)\left(q^2\mathbf{M}_{12}+\ q\mathbf{D}_{12}+\ \mathbf{S}_{12}\right)\right]\beta(q)=\\&\left[\alpha(q)\left(q^2\mathbf{M}_{11}+\ q\mathbf{D}_{11}+\ \mathbf{S}_{11}\right)-\left(q^2\mathbf{M}_{21}+\ q\mathbf{D}_{21}+\ \mathbf{S}_{21}\right)\right]\end{aligned} \tag{4.4}$$

The complex matrix $\beta(q)$ is then found from equation (4.3) as

$$\beta(q)\ =\ G^{-1}(q)\ H(q) \tag{4.5}$$

with the matrices defined as

$$G(q)\ \equiv\ \tilde{\mathbf{L}}_{22}(q)\ -\ \alpha(q)\ \tilde{\mathbf{L}}_{12}(q) \tag{4.6a}$$

and

$$H(q)\ \equiv\ \alpha(q)\ \tilde{\mathbf{L}}_{11}(q)\ -\ \tilde{\mathbf{L}}_{21}(q) \tag{4.6b}$$

Alternatively equations (4.6) can be written explicitly in terms of the partitioned **M**, **D**, and **S** matrices as

$$G(q)\ =\ \left(q^2\mathbf{M}_{22}+\ q\mathbf{D}_{22}+\ \mathbf{S}_{22}\right)-\ \alpha(q)\ \left(q^2\mathbf{M}_{12}+\ q\mathbf{D}_{12}+\ \mathbf{S}_{12}\right) \tag{4.7a}$$

$$H(q)\ =\ \alpha(q)\ \left(q^2\mathbf{M}_{11}+\ q\mathbf{D}_{11}+\ \mathbf{S}_{11}\right)-\left(q^2\mathbf{M}_{21}+\ q\mathbf{D}_{21}+\ \mathbf{S}_{21}\right) \tag{4.7b}$$

Finally reduced system operator, equation (2.12) $\overline{\mathbf{L}}_0(q)$ is

$$\begin{aligned}\overline{\mathbf{L}}_0(q)\ &=\ \tilde{\mathbf{L}}_{11}(q)+\tilde{\mathbf{L}}_{12}(q)\ \beta(q)\\&=\left(q^2\mathbf{M}_{11}+\ q\mathbf{D}_{11}+\ \mathbf{S}_{11}\right)+\left(q^2\mathbf{M}_{12}+\ q\mathbf{D}_{12}+\ \mathbf{S}_{12}\right)\beta(q)\end{aligned} \tag{4.8}$$

from which the reduced system equation can be solved for the response, u_m, of the "master" DOFs

$$\bar{L}_0 \; u_m = \prod \tilde{f} = \tilde{f}_m \tag{4.9}$$

The "Slave" response u_s can then be reconstructed from u_m and $\beta(q)$ according to the equation

$$u_s = \beta(q)\, u_m \tag{4.10}$$

In order to obtain the reduced system operator, equation (4.8), $\bar{L}_0(q)$, it is necessary to determine $\beta(q)$ from equations (4.5, 4.6a and 4.6b). From equation (4.5), it is required to obtain $G^{-1}(q)$ at *every* solution point, q , in order to determine $\beta(q)$ and herein lies the work. The desire is to minimize the work necessary in obtaining $G^{-1}(q)$ at every solution point q, since the inversion of $G(q)$ would involve "slave" DOF by "slave" DOF sized matrix to invert at each point. So rather than determine $G^{-1}(q)$ at <u>*every*</u> solution point, a method has been devised in order to obtain an interpolated $\hat{G}^{-1}(q)$ to be valid within a specified frequency "window".

$$\beta(q) \cong \hat{G}^{-1}(q)\; H(q) \tag{4.11}$$

Two methods, Hermitian interpolation (Section 5) and Lagrangian shape function interpolation (Section6), are proposed in order to interpolate $\hat{G}^{-1}(q)$ at points within a window based on the assumption that $G^{-1}(q)$ is a relatively smoothly varying function of q within the bounds of the defining window. This assumes that any eigenvalues are within the defining window not significantly effect the smooth interpolation of $G^{-1}(q)$.

The interpolation method of $\hat{G}^{-1}(q)$ allows the formulation and calculation of the interpolation or constructor matrices prior to any solution. Hence the main work involved in the matrix inversion and matrix interpolation calculations can be performed as pre-processing steps. This approach can significantly reduce the amount of computation required to generate the reduced system operator, $\bar{L}_0$, and solve the reduced system equation

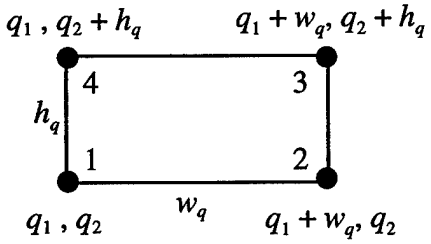
$$\overline{\mathbf{L}}_0 \; \mathbf{u}_m \; = \; \tilde{\mathbf{f}}_m \tag{4.12}$$

since the number of degrees of freedom have been reduced. The reduced computation required for solution of equation (4.12) allows more frequencies to be examined within the interpolation window. Also rather than $\overline{\mathbf{L}}_0$, it is possible to obtain directly the reduced mass, damping and stiffness matrices, such that other or different types of solvers may be use (Appendix C). This would allow the use of third party software to generate the global matrices and to solve the reduced system problem.

5 Hermitian Interpolation Method

5.1 Bi-Cubic Hermitian Interpolation of 2D Rectangular "Window"

Hermitian Interpolations of $\hat{G}^{-1}(q)$ require the determination of $G(q)$ as defined by equation (4.6a) or (4.7a), the inverse, $G^{-1}(q)$, and the first and second derivatives of $G^{-1}(q)$ with respect to q at the four points defining the interpolation window. A "rectangular" window is specified by the four corners, $\overline{q}_j\left(q_1, q_2, w_q, h_q\right)$ where w_q and h_q are the window width and height respectively as defined in complex q-space.



The four vertices of the interpolation window are then

$$\begin{aligned}
\overline{q}_1 &= q_1 + i\, q_2 \\
\overline{q}_2 &= \left(q_1 + w_q\right) + i\, q_2 \\
\overline{q}_3 &= \left(q_1 + w_q\right) + i\left(q_2 + h_q\right) \\
\overline{q}_4 &= q_1 + i\left(q_2 + h_q\right)
\end{aligned}$$

The Hermitian interpolations of the first and second derivatives of $G^{-1}(q)$ with respect to q can be found by the following approximations

$$\frac{\partial[G^{-1}(q)]}{\partial q_j} \cong -G^{-1}(q)\ \frac{\partial G(q)}{\partial q_j}\ G^{-1}(q) \tag{5.1.1}$$

$$\begin{aligned}\frac{\partial^2[G^{-1}(q)]}{\partial q_j \partial q_k} \cong\ & G^{-1}(q)\ \frac{\partial G(q)}{\partial q_j}\ G^{-1}(q)\ \frac{\partial G(q)}{\partial q_k}\ G^{-1}(q) + \\ & G^{-1}(q)\ \frac{\partial G(q)}{\partial q_k}\ G^{-1}(q)\ \frac{\partial G(q)}{\partial q_j}\ G^{-1}(q) - G^{-1}(q)\ \frac{\partial^2 G(q)}{\partial q_j \partial q_k}\ G^{-1}(q)\end{aligned} \tag{5.1.2}$$

where $G^{-1}(q)$ is the matrix inverse of $G(q)$ as defined in equation (4.6a).

From the form of $G(q)$,

$$G(q) = \tilde{\mathbf{L}}_{22}(q) - \alpha(q)\ \tilde{\mathbf{L}}_{12}(q) \tag{5.1.3}$$

the first and second derivatives of $G(q)$ with respect to q_j , i.e. q_1 and q_2 can be found explicitly as

$$\frac{\partial G(q)}{\partial q_j} = \left(\frac{\partial \tilde{\mathbf{L}}_{22}(q)}{\partial q} - \alpha(q)\ \frac{\partial \tilde{\mathbf{L}}_{12}(q)}{\partial q} - \frac{\partial \alpha(q)}{\partial q}\ \tilde{\mathbf{L}}_{12}(q)\right) \frac{\partial q}{\partial q_j} \tag{5.1.4}$$

$$\begin{aligned}\frac{\partial^2 G(q)}{\partial q_j \partial q_k} = & \left(\frac{\partial^2 \tilde{\mathbf{L}}_{22}(q)}{\partial q^2} - \alpha(q)\ \frac{\partial^2 \tilde{\mathbf{L}}_{12}(q)}{\partial q^2} - \frac{\partial \alpha(q)}{\partial q}\ \frac{\partial \tilde{\mathbf{L}}_{12}(q)}{\partial q} - \right. \\ & \left. \frac{\partial \alpha(q)}{\partial q}\ \frac{\partial \tilde{\mathbf{L}}_{12}(q)}{\partial q} - \frac{\partial^2 \alpha(q)}{\partial q^2}\ \tilde{\mathbf{L}}_{12}(q)\right) \frac{\partial q}{\partial q_j}\ \frac{\partial q}{\partial q_k}\end{aligned} \tag{5.1.5}$$

$$= \left(\frac{\partial^2 \tilde{\mathbf{L}}_{22}(q)}{\partial q^2} - \alpha(q)\frac{\partial^2 \tilde{\mathbf{L}}_{12}(q)}{\partial q^2} - 2\ \frac{\partial \alpha(q)}{\partial q}\frac{\partial \tilde{\mathbf{L}}_{12}(q)}{\partial q} - \frac{\partial^2 \alpha(q)}{\partial q^2}\tilde{\mathbf{L}}_{12}(q)\right) \frac{\partial q}{\partial q_j}\frac{\partial q}{\partial q_k}$$

Also from the definition of $q = q_1 + iq_2$ the derivatives $\frac{\partial q}{\partial q_j}$ can easily be found as

$$\frac{\partial q}{\partial q_1} = 1 \qquad \text{and} \qquad \frac{\partial q}{\partial q_2} = i$$

Having obtained the necessary matrix terms, it is now possible to formulate the matrices needed to construct the interpolated $\hat{G}^{-1}(q)$.

The real interpolation coefficient matrix $B(\overline{q}_j)$ is calculated from the matrices $C_j(\overline{q}_j)$ at the corners of the rectangular window, $\overline{q}_j$ as

$$B(\overline{q}_j) = B(q_1, q_2, w_q, h_q) = \begin{pmatrix} C_1(q_1, q_2) \\ C_2(q_1 + w_q, q_2) \\ C_3(q_1 + w_q, q_2 + h_q) \\ C_4(q_1, q_2 + h_q) \end{pmatrix} \tag{5.1.6}$$

from which the matrix inverse $B^{-1}(\overline{q}_j)$ is found. The C_j interpolation coefficient matrices are obtained at each of the points $\overline{q}_j$ from

$$C_j(\overline{q}_j) = \begin{pmatrix} 1 & q_1 & q_1^2 & q_1^3 & q_2 & q_1 q_2 & q_1^2 q_2 & q_1^3 q_2 \\ 0 & 1 & 2q_1 & 3q_1^2 & 0 & q_2 & 2q_1 q_2 & 3q_1^2 q_2 \\ 0 & 0 & 0 & 0 & 1 & q_1 & q_1^2 & q_1^3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2q_1 & 3q_1^2 \end{pmatrix} \tag{5.1.7}$$

$$\begin{pmatrix} q_2^2 & q_1 q_2^2 & q_1^2 q_2^2 & q_1^3 q_2^2 & q_2^3 & q_1 q_2^3 & q_1^2 q_2^3 & q_1^3 q_2^3 \\ 0 & q_2^2 & 2q_1 q_2^2 & 3q_1^2 q_2^2 & 0 & q_2^3 & 2q_1 q_2^3 & 3q_1^2 q_2^3 \\ 2q_2 & 2q_1 q_2 & 2q_1^2 q_2 & 2q_1^3 q_2 & 3q_2^2 & 3q_1 q_2^3 & 3q_1^2 q_2^2 & 3q_1^3 q_2^2 \\ 0 & 2q_2 & 4q_1 q_2 & 6q_1^2 q_2 & 0 & 3q_2^2 & 6q_1 q_2^2 & 9q_1^2 q_2^2 \end{pmatrix}$$

Defining the matrix $u_j(\overline{q}_j)$ at a specific point $\overline{q}_j$ as

$$u_j(q_1,q_2)=\begin{pmatrix} G^{-1}(q_1,q_2) \\ \dfrac{\partial G^{-1}(q_1,q_2)}{\partial q_1} \\ \dfrac{\partial G^{-1}(q_1,q_2)}{\partial q_2} \\ \dfrac{\partial^2 G^{-1}(q_1,q_2)}{\partial q_1 \partial q_2} \end{pmatrix} \tag{5.1.8}$$

the full $U(\bar{q}_j)$ matrix at all the interpolation window points $\bar{q}_j, j = 1,2,3,4$ is defined as

$$U(\bar{q})=\begin{pmatrix} G^{-1}(q_1,q_2) \\ \dfrac{\partial \hat{G}^{-1}(q_1,q_2)}{\partial q_1} \\ \dfrac{\partial \hat{G}^{-1}(q_1,q_2)}{\partial q_2} \\ \dfrac{\partial^2 \hat{G}^{-1}(q_1,q_2)}{\partial q_1 \partial q_2} \\ G^{-1}(q_1+w,q_2) \\ \dfrac{\partial \hat{G}^{-1}(q_1+w,q_2)}{\partial q_1} \\ \dfrac{\partial \hat{G}^{-1}(q_1+w,q_2)}{\partial q_2} \\ \dfrac{\partial^2 \hat{G}^{-1}(q_1+w,q_2)}{\partial q_1 \partial q_2} \\ G^{-1}(q_1+w,q_2+h) \\ \dfrac{\partial \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_1} \\ \dfrac{\partial \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_2} \\ \dfrac{\partial^2 \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_1 \partial q_2} \\ G^{-1}(q_1+w,q_2+h) \\ \dfrac{\partial \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_1} \\ \dfrac{\partial \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_2} \\ \dfrac{\partial^2 \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_1 \partial q_2} \end{pmatrix} \tag{5.1.9}$$

Finally $W(\bar{q}_j)$, the interpolation constructor matrices, can be found as

$$W(\bar{q}_j) = B^{-1}(\bar{q}_j) \circ U(\bar{q}_j) \tag{5.1.10}$$

NOTE: Equation 5.1.10 may be written explicitly as

$$W_i(\bar{q}_k) = \sum_{j=1}^{16} B_{ij}^{-1}(\bar{q}_k)\ U_j(\bar{q}_k)$$

where i = 1,16 and k = 1,2,3,4, i.e. the interpolation points. So, for example, the first term would be

$$\begin{aligned} W_1(\bar{q}_k) = {} & B_{1,1}^{-1}\ G^{-1}(\bar{q}_1) + B_{1,2}^{-1}\frac{\partial \hat{G}^{-1}(\bar{q}_1)}{\partial q_1} + B_{1,3}^{-1}\frac{\partial \hat{G}^{-1}(\bar{q}_1)}{\partial q_2} + B_{1,4}^{-1}\frac{\partial^2 \hat{G}^{-1}(\bar{q}_1)}{\partial q_2 \partial q_1} + \\ & B_{1,5}^{-1}\ G^{-1}(\bar{q}_2) + B_{1,6}^{-1}\frac{\partial \hat{G}^{-1}(\bar{q}_2)}{\partial q_1} + B_{1,7}^{-1}\frac{\partial \hat{G}^{-1}(\bar{q}_2)}{\partial q_2} + B_{1,8}^{-1}\frac{\partial^2 \hat{G}^{-1}(\bar{q}_2)}{\partial q_2 \partial q_1} + \\ & B_{1,9}^{-1}\ G^{-1}(\bar{q}_3) + B_{1,10}^{-1}\frac{\partial \hat{G}^{-1}(\bar{q}_3)}{\partial q_1} + B_{1,11}^{-1}\frac{\partial \hat{G}^{-1}(\bar{q}_3)}{\partial q_2} + B_{1,12}^{-1}\frac{\partial^2 \hat{G}^{-1}(\bar{q}_3)}{\partial q_2 \partial q_1} + \\ & B_{1,13}^{-1}\ G^{-1}(\bar{q}_4) + B_{1,14}^{-1}\frac{\partial \hat{G}^{-1}(\bar{q}_4)}{\partial q_1} + B_{1,15}^{-1}\frac{\partial \hat{G}^{-1}(\bar{q}_4)}{\partial q_2} + B_{1,16}^{-1}\frac{\partial^2 \hat{G}^{-1}(\bar{q}_4)}{\partial q_2 \partial q_1} \end{aligned}$$

where B_{ij}^{-1} are the *individual* B matrix inverse terms.


It is important to note that $W(\bar{q}_j)$ and $U(\bar{q}_j)$ are 16 COMPLEX "Slave" DOF by "Slave" DOF matrices and $B^{-1}(\bar{q}_j)$ is a 16 by 16 REAL matrix. The $W(\bar{q}_j)$ interpolation constructor matrices <u>MUST</u> be stored for the further solution.

Having found $W(\bar{q}_j)$ for the rectangular window at the points $\bar{q}_j$ the $\hat{G}^{-1}(\hat{q})$ of equation (4.11) may now be interpolated for a given value of $\hat{q} = f(\hat{q}_1, \hat{q}_2)$ within the window as

$$\hat{G}^{-1}(\hat{q}) \cong C_0(\hat{q}_1, \hat{q}_2)\ W(\bar{q}_j) \tag{5.1.11}$$

where

$$C_0(\hat{q}_1,\hat{q}_2) = \left(1 \quad \hat{q}_1 \quad \hat{q}_1^2 \quad \hat{q}_1^3 \quad \hat{q}_2 \quad \hat{q}_1\hat{q}_2 \quad \hat{q}_1^2\hat{q}_2 \quad \hat{q}_1^3\hat{q}_2 \right.$$

$$\left. \hat{q}_2^2 \quad \hat{q}_1\hat{q}_2^2 \quad \hat{q}_1^2\hat{q}_2^2 \quad \hat{q}_1^3\hat{q}_2^2 \quad \hat{q}_2^3 \quad \hat{q}_1\hat{q}_2^3 \quad \hat{q}_1^2\hat{q}_2^3 \quad \hat{q}_1^3\hat{q}_2^3\right) \tag{5.1.12}$$

now $\beta(q)$ (equation (4.5)) at the value of $\hat{q} = f(\hat{q}_1,\hat{q}_2)$ within the window is then approximated as

$$\beta(\hat{q}) \cong \hat{G}^{-1}(\hat{q})\ H(\hat{q}) \tag{5.1.13}$$

and so equation (4.8) can be evaluated at $\hat{q}$ in order to find the reduced system operator

$$\overline{\mathbf{L}}_0(\hat{q}) = \tilde{\mathbf{L}}_{11}(\hat{q}) + \tilde{\mathbf{L}}_{12}(\hat{q})\ \beta(\hat{q}) \tag{5.1.14}$$

from which the reduced system equation can be solved for the response u_m

$$\overline{\mathbf{L}}_0\ u_m = \tilde{f}_m \tag{5.1.15}$$

It should be noted that the matrix $\beta(q)$ is a COMPLEX "Slave" DOF by "Master" DOF matrix and MUST be stored if the "Slave" response is to be reconstructed as

$$u_s = \beta(\hat{q})\, u_m \tag{5.1.16}$$

5.2 Hermitian Interpolation of 1D Linear "Window"

5.2.1 End Point Hermitian Interpolation

If there is a desire to confine the analysis to a frequency range (line) which is either along the "Real" or "Imaginary" axes, the above equations may be greatly simplified. The criteria are for a line along the "Real" axis,

$$\omega_1 \neq 0\ ,\ \omega_2 = 0\ ,\ w_\omega \neq 0\ ,\ h_\omega = 0$$

and for a line along the "Imaginary" axis,

$$\omega_1 = 0\ ,\ \omega_2 \neq 0\ ,\ w_\omega = 0\ ,\ h_\omega \neq 0$$

Here the interpolation points of a frequency range or line, $\overline{\omega}_j(\omega_1,\omega_2,w_\omega,h_\omega) = \omega_0,\omega_e$ are defined as the line end points ; $\omega_0 = \omega_1 + i\ \omega_2$ is the origin or beginning of the line and *either* $\omega_e = (\omega_1 + w_\omega) + i\ \omega_2$ is the end point of the line for the case where $w_\omega \neq 0$; $h_\omega = 0$, i.e. along the "real" axis *or* $\omega_e = \omega_1 + i\ (\omega_2 + h_\omega)$ for the case where $w_\omega = 0$; $h_\omega \neq 0$, i.e. along the "imaginary" axis.

The $\hat{G}^{-1}(\hat{\omega})$ is now simply be approximated at any point along the line, $\hat{\omega} = f(\hat{\omega}_1,\hat{\omega}_2)$, as

$$\hat{G}^{-1}(\hat{\omega}) \cong W_1(\overline{\omega}) + \hat{\omega}\ W_2(\overline{\omega}) + \hat{\omega}^2\ W_3(\overline{\omega}) + \hat{\omega}^3\ W_4(\overline{\omega}) \tag{5.2.1}$$

The complex constructor matrices $W_j(\overline{\omega})$ are determined similarly to the 2D case, for the line end points, $\overline{\omega}_j$, as

$$\begin{pmatrix} W_1 \\ W_2 \\ W_3 \\ W_4 \end{pmatrix} = B^{-1}(\omega_0,\omega_e) \circ \begin{pmatrix} G^{-1}(\omega_0) \\ \dfrac{\partial G^{-1}(\omega_0)}{\partial \omega} \\ G^{-1}(\omega_e) \\ \dfrac{\partial G^{-1}(\omega_e)}{\partial \omega} \end{pmatrix} \tag{5.2.2}$$

and the real interpolation coefficient matrix $B^{-1}(\omega_0,\omega_e)$ is the matrix inverse of

$$B(\omega_0,\omega_e) = B(\omega_1,\omega_2,w_\omega,h_\omega) = \begin{pmatrix} 1 & \omega_0 & \omega_0^2 & \omega_0^3 \\ 0 & 1 & 2\omega_0 & 3\omega_0^2 \\ 1 & \omega_e & \omega_e^2 & \omega_e^3 \\ 0 & 1 & 2\omega_e & 3\omega_e^2 \end{pmatrix} \tag{5.2.3}$$

As in the 2D "window" case above, the Hermitian interpolations of the first derivative of $G^{-1}(q)$ with respect to q are as above in equation (5.1.1)

$$\frac{\partial[G^{-1}(q)]}{\partial q_j} \equiv -G^{-1}(q)\,\frac{\partial G(q)}{\partial q_j}\,G^{-1}(q) \tag{5.2.4}$$

where first derivative of $\frac{\partial G^{-1}(\overline{\omega}_j)}{\partial \omega_i}$ may be obtained from equation (5.1.4) via the chain rule as

$$\frac{\partial G(q)}{\partial q_j}\frac{\partial q_j}{\partial \omega_j} = \left(\frac{\partial \tilde{\mathbf{L}}_{22}(q)}{\partial q} - \alpha(q)\,\frac{\partial \tilde{\mathbf{L}}_{12}(q)}{\partial q} - \frac{\partial \alpha(q)}{\partial q}\,\tilde{\mathbf{L}}_{12}(q)\right)\frac{\partial q}{\partial q_j}\frac{\partial q_j}{\partial \omega_j} \tag{5.2.5}$$

As before from the definition $q = q_1 + iq_2$ i.e. $q \equiv i\omega$, the respective derivatives are

$$\frac{\partial q}{\partial \omega_1} = i \quad \text{and} \quad \frac{\partial q}{\partial \omega_2} = -1 \tag{5.2.6}$$

The cases for a line *parallel* the "Real" axis, $\omega_1 \neq 0$, $\omega_2 \neq 0$, $w_\omega \neq 0$, $h_\omega = 0$ or for a line *parallel* the "Imaginary" axis, $\omega_1 \neq 0$, $\omega_2 \neq 0$, $w_\omega = 0$, $h_\omega \neq 0$, the frequency can be defined as $\omega = |\omega|\, e^{i\,\theta} + a$ where $|\omega|$ is a frequency magnitude, θ is a real fixed value of either zero or $\frac{\pi}{2}$ and a is a complex origin offset. If ω parallel to either the real or imaginary axis then $a \neq 0$. So for ω parallel to the real axis for $\theta = 0$ and $a = 0 + ia_2$, i.e. ω_j is offset from the real axis by ia_2 or for $\theta = \frac{\pi}{2}$ and $a = a_1 + i0$, i.e. ω_j is offset from the imaginary axis by a_1. Then the values for ω_0 and ω_e represent magnitudes as defined as

$$|\omega_j| = \sqrt{(\omega_j - a)^*(\omega_j - a)} \tag{5.2.7}$$

where * represents the complex conjugate.

Note for the cases where ω is on an axis $a = 0$; if $\theta = 0$; i.e. ω along to the real axis or if $\theta = \frac{\pi}{2}$; i.e. ω along to the imaginary axis, the magnitude is simply

$$|\omega_j| = \sqrt{(\omega_j)^*(\omega_j)} = \omega_j \tag{5.2.8}$$

where * represents the complex conjugate.

5.2.2 Generalized 1D Hermitian Interpolation

The generalized 1D Hermitian Interpolation allows more than two interpolation (inversion) points and can utilize higher order Hermitian Interpolations. However, here, we shall restrict the analysis to 1st order Hermitian Interpolation with up to 4 interpolation points. Also the equations shall be derived in terms of the generalized variable $q = i\omega$.

The $\hat{G}^{-1}(\hat{q})$ may be approximated at any point on a line, $\hat{q} = f(\hat{q}_1, \hat{q}_2)$, as

$$\hat{G}^{-1}(\hat{q}) = W_1(\bar{q}) + \hat{q}\ W_2(\bar{q}) + \hat{q}^2\ W_3(\bar{q}) + \cdots + \hat{q}^{[(Order+1)N]-1}\ W_{(Order+1)N}(\bar{q}) \tag{5.2.9}$$

It should be noted that for simplicity, interpolation will be restricted to a line either *on* the "Real" axis *or* a line *on* the "Imaginary" axis. So from the definition,

$$q = i\omega = i(\omega_1 + i\omega_2) \qquad \text{then} \qquad q = q_1 + iq_2 = -\omega_2 + i\omega_1 .$$

Therefore, for real frequencies, ω_1, *on* the "Real" frequency axis

$$\omega_1 \neq 0,\ \omega_2 = 0,\ w_\omega \neq 0,\ h_\omega = 0 \quad \text{and so}$$
$$q_1 = 0,\ q_2 \neq 0,\ w_q = 0,\ h_q \neq 0$$

and for imaginary frequencies, ω_2, *on* the "Imaginary" frequency axis

$$\omega_1 = 0,\ \omega_2 \neq 0,\ w_\omega = 0,\ h_\omega \neq 0 \quad \text{and so}$$
$$q_1 \neq 0,\ q_2 = 0,\ w_q \neq 0,\ h_q = 0$$

(See Appendix B)

For 1st order Hermitian Interpolation, there will be $2N$, i.e. $(Order+1)N$, W matrices needed for the approximation of $\hat{G}^{-1}(\hat{q})$ in equation (5.2.6). These matrices may be calculated as

$$\begin{pmatrix} W_1 \\ W_2 \\ \vdots \\ W_{2N} \end{pmatrix} = B^{-1}(\bar{q}) \circ \begin{pmatrix} G^{-1}(\bar{q}_1) \\ \dfrac{\partial G^{-1}(\bar{q}_1)}{\partial q} \\ G^{-1}(\bar{q}_2) \\ \dfrac{\partial G^{-1}(\bar{q}_3)}{\partial q} \\ \vdots \\ G^{-1}(\bar{q}_N) \\ \dfrac{\partial G^{-1}(\bar{q}_N)}{\partial q} \end{pmatrix} \tag{5.2.10}$$

NOTE: 1st order Hermitian Interpolation requires only the first derivative of $G^{-1}(q)$ with respect to q. Higher order Hermitian Interpolation would require higher order derivatives.

The interpolation coefficient matrices, $B(q)$, for different number of interpolation points, N = 2 , 3 , or 4, and for 1st order Hermitian Interpolations may be written explicitly as

For N = 2 $$B(\bar{q}) = \begin{pmatrix} 1 & \bar{q}_1 & \bar{q}_1^2 & \bar{q}_1^3 \\ 0 & 1 & 2\bar{q}_1 & 3\bar{q}_1^2 \\ 1 & \bar{q}_2 & \bar{q}_2^2 & \bar{q}_2^3 \\ 0 & 1 & 2\bar{q}_2 & 3\bar{q}_2^2 \end{pmatrix} \tag{5.2.11a}$$

For N = 3 $$B(\bar{q}) = \begin{pmatrix} 1 & \bar{q}_1 & \bar{q}_1^2 & \bar{q}_1^3 & \bar{q}_1^4 & \bar{q}_1^5 \\ 0 & 1 & 2\bar{q}_1 & 3\bar{q}_1^2 & 4\bar{q}_1^3 & 5\bar{q}_1^4 \\ 1 & \bar{q}_2 & \bar{q}_2^2 & \bar{q}_2^3 & \bar{q}_2^4 & \bar{q}_2^5 \\ 0 & 1 & 2\bar{q}_2 & 3\bar{q}_2^2 & 4\bar{q}_2^3 & 5\bar{q}_2^4 \\ 1 & \bar{q}_3 & \bar{q}_3^2 & \bar{q}_3^3 & \bar{q}_3^4 & \bar{q}_3^5 \\ 0 & 1 & 2\bar{q}_3 & 3\bar{q}_3^2 & 4\bar{q}_3^3 & 5\bar{q}_3^4 \end{pmatrix} \tag{5.2.11b}$$

For N = 4 $\quad B(\bar{q}) = \begin{pmatrix} 1 & \bar{q}_1 & \bar{q}_1^2 & \bar{q}_1^3 & \bar{q}_1^4 & \bar{q}_1^5 & \bar{q}_1^6 & \bar{q}_1^7 \\ 0 & 1 & 2\bar{q}_1 & 3\bar{q}_1^2 & 4\bar{q}_1^3 & 5\bar{q}_1^4 & 6\bar{q}_1^5 & 7\bar{q}_1^6 \\ 1 & \bar{q}_2 & \bar{q}_2^2 & \bar{q}_2^3 & \bar{q}_2^4 & \bar{q}_2^5 & \bar{q}_2^6 & \bar{q}_2^7 \\ 0 & 1 & 2\bar{q}_2 & 3\bar{q}_2^2 & 4\bar{q}_2^3 & 5\bar{q}_2^4 & 6\bar{q}_2^5 & 7\bar{q}_2^6 \\ 1 & \bar{q}_3 & \bar{q}_3^2 & \bar{q}_3^3 & \bar{q}_3^4 & \bar{q}_3^5 & \bar{q}_3^6 & \bar{q}_3^7 \\ 0 & 1 & 2\bar{q}_3 & 3\bar{q}_3^2 & 4\bar{q}_3^3 & 5\bar{q}_3^4 & 6\bar{q}_3^5 & 7\bar{q}_3^6 \\ 1 & \bar{q}_4 & \bar{q}_4^2 & \bar{q}_4^3 & \bar{q}_4^4 & \bar{q}_4^5 & \bar{q}_4^6 & \bar{q}_4^7 \\ 0 & 1 & 2\bar{q}_4 & 3\bar{q}_4^2 & 4\bar{q}_4^3 & 5\bar{q}_4^4 & 6\bar{q}_4^5 & 7\bar{q}_4^6 \end{pmatrix}$ (5.2.11c)

As above in the 2D "window" case above, the Hermitian interpolations of the first derivative of $G^{-1}(q)$ with respect to q are as above in equation (5.1.1)

$$\frac{\partial\left[G^{-1}(q)\right]}{\partial q_j} \cong -G^{-1}(q)\,\frac{\partial G(q)}{\partial q_j}\,G^{-1}(q) \tag{5.2.12}$$

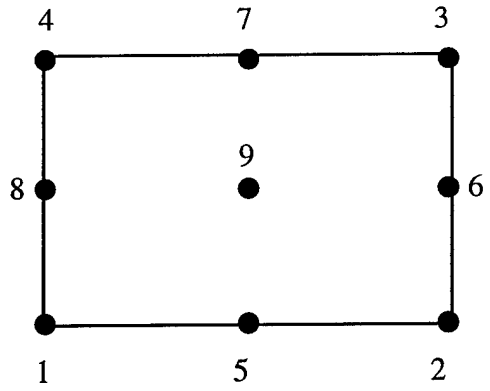
The first derivative of $\dfrac{\partial G^{-1}(\bar{q}_j)}{\partial q}$ may be obtained from equation (5.1.4) and from the definition of $q = q_1 + iq_2$ with the derivatives $\dfrac{\partial q}{\partial q_j}$ as $\dfrac{\partial q}{\partial q_1} = 1$ and $\dfrac{\partial q}{\partial q_2} = i$ and so

$$\frac{\partial G(\bar{q})}{\partial q_j} = \left(\frac{\partial \tilde{\mathbf{L}}_{22}(\bar{q})}{\partial q} - \alpha(\bar{q})\,\frac{\partial \tilde{\mathbf{L}}_{12}(\bar{q})}{\partial q} - \frac{\partial \alpha(\bar{q})}{\partial q}\,\tilde{\mathbf{L}}_{12}(\bar{q})\right)\frac{\partial \bar{q}}{\partial q_j} \tag{5.2.13}$$

6 Lagrangian Interpolation Method

6.1 2D Interpolation using Rectangular Lagrangian Shape Functions

Isoparametric 2D quadratic shape functions of a rectangular "element" can be used to interpolate any point within the element. A rectangular FEM element with 4 to 9 nodes having the following node numbering is used to represent the frequency window to be interpolated.



The 2 dimensional shape functions, $h_i(r,s)$, in isoparametric coordinates (r,s) for any nodal configuration are defined in the table below.

Table 6.1

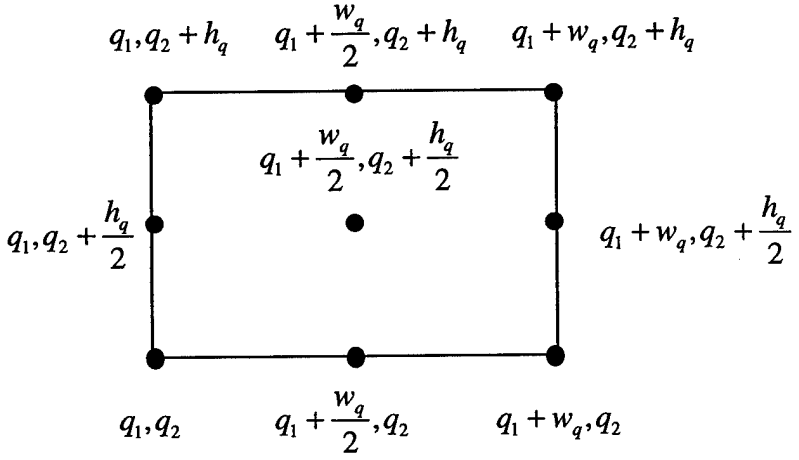
Include only if Node i is Defined

		i = 5	i = 6	i = 7	i = 8	i = 9
h_1	0.25 (1-r) (1-s)	-0.5 h_5			-0.5 h_8	-0.25 h_9
h_2	0.25 (1+r) (1-s)	-0.5 h_5	-0.5 h_6			-0.25 h_9
h_3	0.25 (1+r) (1+s)		-0.5 h_6	-0.5 h_7		-0.25 h_9
h_4	0.25 (1-r) (1+s)			-0.5 h_7	-0.5 h_8	-0.25 h_9
h_5	0.5 (1-r^2) (1-s)					-0.5 h_9
h_6	0.5 (1+r) (1-s^2)					-0.5 h_9
h_7	0.5 (1-r^2) (1+s)					-0.5 h_9
h_8	0.5 (1-r) (1-s^2)					-0.5 h_9
h_9	(1-r^2) (1-s^2)					

From this table, the 2D shape functions, $h_i(r,s)$, for the represented "element" above, that is the nine nodal points, can be written explicitly as

$$
\begin{aligned}
h_9(r,s) &= \left(1-r^2\right)\left(1-s^2\right) \\
h_8(r,s) &= 0.5\,(1-r)\left(1-s^2\right) - 0.5\,h_9(r,s) \\
h_7(r,s) &= 0.5\left(1-r^2\right)(1-s) - 0.5\,h_9(r,s) \\
h_6(r,s) &= 0.5\,(1+r)\left(1-s^2\right) - 0.5\,h_9(r,s) \\
h_5(r,s) &= 0.5\left(1-r^2\right)(1-s) - 0.5\,h_9(r,s) \\
h_4(r,s) &= 0.25\,(1-r)(1+s) - 0.5\,h_7(r,s) - 0.5\,h_8(r,s) - 0.25\,h_9(r,s) \\
h_3(r,s) &= 0.25\,(1+r)(1+s) - 0.5\,h_6(r,s) - 0.5\,h_7(r,s) - 0.25\,h_9(r,s) \\
h_2(r,s) &= 0.25\,(1+r)(1-s) - 0.5\,h_5(r,s) - 0.5\,h_6(r,s) - 0.25\,h_9(r,s) \\
h_1(r,s) &= 0.25\,(1-r)(1-s) - 0.5\,h_5(r,s) - 0.5\,h_8(r,s) - 0.25\,h_9(r,s)
\end{aligned}
\qquad (6.1.1)
$$

Defining q_1, q_2 as the origin of the rectangular window to be interpolated, w_q as the window width and h_q as the window height, the nodal point coordinate values are then defined as in the figure below.



The matrix $G(q)$ as in equation (4.6a) and (4.7a)

$$
\begin{aligned}
G(q) &= \tilde{\mathbf{L}}_{22}(q) - \alpha(q)\,\tilde{\mathbf{L}}_{12}(q) \\
&= \left(q^2\mathrm{M}_{22} + q\mathrm{D}_{22} + \mathrm{S}_{22}\right) - \alpha(q)\left(q^2\mathrm{M}_{12} + q\mathrm{D}_{12} + \mathrm{S}_{12}\right)
\end{aligned}
\qquad (6.1.2)
$$

must be evaluated at each of the nine nodal points, $\overline{q}_j$ (or fewer if less interpolation points are used) so the inverse matrix $G^{-1}(\overline{q}_j)$ can then be obtained at each of the interpolation points and saved for further use. It should be noted that there will be a maximum of 9 COMPLEX "Slave" DOF by "Slave" DOF matrices of the respective $G^{-1}(\overline{q}_j)$ at the nine nodal points stored for the further solution.

For a given value of $\hat{q} = f(\hat{q}_1,\hat{q}_2)$ within the window i.e. $q_1 \le \hat{q}_1 \le q_1 + w_q$ and $q_2 \le \hat{q}_2 \le q_2 + h_q$, the corresponding isoparametric coordinates (r,s) can be found from the following relationships

$$\hat{r} = \frac{-2\left(0.5\ w_q + q_1 - \hat{q}_1\right)}{w_q} \tag{6.1.3a}$$

$$\hat{s} = \frac{-2\left(0.5\ h_q + q_2 - \hat{q}_2\right)}{h_q} \tag{6.1.3b}$$

The value of $\hat{G}^{-1}(\hat{q})$ can be interpolated by the shape functions evaluated at $\hat{q} = f(\hat{q}_1,\hat{q}_2)$ from the isoparametric coordinates $(\hat{r},\hat{s})$ and the $G^{-1}(\overline{q}_j)$ matrices previously calculated at each of the "element" or window interpolation points by

$$\hat{G}^{-1}(\hat{q}) = \sum_{i=1}^{9} h_i(\hat{r},\hat{s})\ G_i^{-1}(\overline{q}) \tag{6.1.4}$$

The matrix $\beta(\hat{q})$ is then approximated as

$$\beta(\hat{q}) \cong \hat{G}^{-1}(\hat{q})H(\hat{q}) \tag{6.1.5}$$

This is now use to determine the reduced operator $\tilde{\mathbf{L}}_0(\hat{q})$ as before

$$\begin{aligned}\tilde{\mathbf{L}}_0(\hat{q}) &= \tilde{\mathbf{L}}_{11}(\hat{q}) + \tilde{\mathbf{L}}_{12}(\hat{q})\ \beta(\hat{q}) \\ &= \left(\hat{q}^2\mathrm{M}_{11} + \hat{q}\mathrm{D}_{11} + \mathrm{S}_{11}\right) + \left(\hat{q}^2\mathrm{M}_{12} + \hat{q}\mathrm{D}_{12} + \mathrm{S}_{12}\right)\ \beta(\hat{q})\end{aligned} \tag{6.1.6}$$

The reduced system equation can be solved, as before, for the response $\mathbf{u}_m$

$$\tilde{L}_0 \; u_m \; = \; \prod \tilde{f} \; = \; \tilde{f}_m$$

and "Slave" response u_s can be reconstructed by

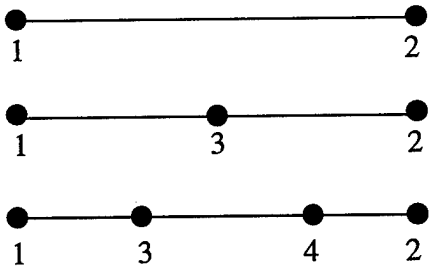
$$u_s \; = \; \beta(\hat{q}) \, u_m \tag{6.1.7}$$

The matrix $\beta(\hat{q})$ is a COMPLEX "Slave" DOF by "Master" DOF matrix and MUST be stored if the "Slave" response is to be reconstructed.

6.2 Lagrangian Interpolation of 1D Linear "Window"

6.2.1 Fixed Point Lagrangian Shape Functions

Isoparametric 1D shape functions of a linear "element" can be used to interpolate any point along a selected line. A 1D FEM linear element with 2 to 4 nodes having the following node numbering is used to represent a line either along or parallel to the "Real" *or* the "Imaginary" frequency axes.



The 1 dimensional shape functions, $h_i(r)$, in isoparametric coordinate r for any nodal configuration are defined in the table below.

Table 6.2

		Include Only if Node 3 is Present	Include Only if Nodes 3 and 4 are Present
h_1	$0.5\ (1-r)$	$-0.5\ (1-r^2)$	$+\frac{1}{16}(-9r^3+r^2+9r-1)$
h_2	$0.5\ (1+r)$	$-0.5\ (1-r^2)$	$+\frac{1}{16}(9r^3+r^2-9r-1)$
h_3	$(1-r^2)$		$+\frac{1}{16}(27r^3+7r^2-27r-7)$
h_4	$\frac{1}{16}(-27r^3-9r^2+27r+9)$		

NOTE: For the 3 node shape functions, point 3 is fixed at a value of one-half. For the 4 node shape functions, point 3 is fixed at one-third and point 4 is fixed at two-thirds.

From Table 6.2, the 1D shape functions, $h_i(r)$, for a 4 node line interpolation , can be written explicitly as

$$\begin{aligned} h_4(r) &= \frac{1}{16}(-27r^3 - 9r^2 + 27r - 9) \\ h_3(r) &= (1-r^2) + \frac{1}{16}(27r^3 + 7r^2 - 27r - 7) \\ h_2(r) &= 0.5(1+r) - 0.5(1-r^2) + \frac{1}{16}(9r^3 + r^2 - 9r - 1) \\ h_1(r) &= 0.5(1-r) - 0.5(1-r^2) + \frac{1}{16}(-9r^3 + r^2 + 9r - 1) \end{aligned} \qquad (6.2.1)$$

For a given value of $\hat{q} = f(\hat{q}_1, \hat{q}_2)$ along a line i.e. $q_1 \le \hat{q}_1 \le q_1 + w_q$ if $h_q = 0$ or $q_2 \le \hat{q}_2 \le q_2 + h_q$ if $w_q = 0$, the corresponding isoparametric coordinate r can be found from the relationship in equation (6.1.3a) as *either*

$$\hat{r} = \frac{-2\left(0.5\ w_q + q_1 - \hat{q}_1\right)}{w_q} \qquad \text{if } h_q = 0 \qquad (6.2.2a)$$

or

$$\hat{r} = \frac{-2\left(0.5\ h_q + q_2 - \hat{q}_2\right)}{h_q} \qquad \text{if } w_q = 0 \qquad (6.2.2b)$$

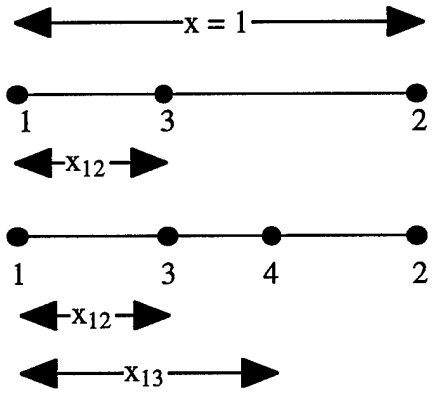
The value of $\hat{G}^{-1}(\hat{q})$ can be interpolated by the shape functions evaluated at $\hat{q} = f(\hat{q}_1, \hat{q}_2)$ from the isoparametric coordinate $\hat{r}$ and the $G^{-1}(\overline{q})$ matrices previously calculated at each of the line interpolation points by

$$\hat{G}^{-1}(\hat{q}) = \sum_{i=1}^{4} h_i(\hat{r})\ G_i^{-1}(\overline{q}) \tag{6.2.3}$$

$\beta(\hat{q})$ is then approximated as before as in equation (6.1.5)

6.2.2 Variable Point Lagrangian Shape Functions

If it is desired to have variable locations for the 1D-shape functions, the following forms are used.



For a given value of $\hat{q} = f(\hat{q}_1, \hat{q}_2)$ along a line i.e. $q_1 \le \hat{q} \le q_1 + w_q$, if $h_q = 0$ or $q_2 \le \hat{q} \le q_2 + h_q$, if $w_q = 0$, the term Δx is defined as

$$\Delta x = \frac{(\hat{q}_1 - q_1)}{w_q} \qquad \text{if } h_q = 0 \tag{6.2.5a}$$

or

$$\Delta x = \frac{(\hat{q}_2 - q_2)}{h_q} \qquad \text{if } w_q = 0 \tag{6.2.5b}$$

Now the corresponding 3 point quadratic shape functions are written as

$$
\begin{aligned}
h_1(q) &= 1 + \frac{x + x_{12}}{x\ x_{12}}\,\Delta x + \frac{1}{x\ x_{12}}\,\Delta x^2 \\
h_2(q) &= \frac{x_{12}}{x(x - x_{12})}\,\Delta x + \frac{1}{x(x - x_{12})}\,\Delta x^2 \\
h_3(q) &= \frac{x}{x_{12}(x - x_{12})}\,\Delta x - \frac{1}{x_{12}(x - x_{12})}\,\Delta x^2
\end{aligned}
\qquad (6.2.6)
$$

and the 4 point cubic shape functions are

$$
\begin{aligned}
h_1(q) &= 1 - \frac{\big((x\ x_{12}) + (x\ x_{13}) + (x_{12}\ x_{13})\big)}{x\ x_{12}\ x_{13}}\,\Delta x - \frac{(-x - x_{12} - x_{13})}{x\ x_{12}\ x_{13}}\,\Delta x^2 - \frac{1}{x\ x_{12}\ x_{13}}\,\Delta x^3 \\
h_2(q) &= \frac{x_{12}\ x_{13}}{x(x - x_{12})(x - x_{13})}\,\Delta x - \frac{(-x_{12} - x_{13})}{x(x - x_{12})(x - x_{13})}\Delta x^2 + \frac{1}{x(x - x_{12})(x - x_{13})}\,\Delta x^3
\end{aligned}
\qquad (6.2.7)
$$

$$
h_3(q) = -\frac{x\ x_{13}}{x_{12}(x - x_{12})(x_{12} - x_{13})}\Delta x - \frac{(-x - x_{13})}{x_{12}(x - x_{12})(x_{12} - x_{13})}\,\Delta x^2 - \frac{1}{x_{12}(x - x_{12})(x - x_{13})}\,\Delta x^3
$$

$$
h_4(q) = -\frac{x\ x_{12}}{x_{13}(x - x_{13})(-x_{12} + x_{13})}\Delta x - \frac{x + x_{12}}{x_{13}(x - x_{13})(x_{12} - x_{13})}\,\Delta x^2 + \frac{1}{x_{13}(x - x_{13})(x_{12} - x_{13})}\Delta x^3
$$

Here x is defined as unity, i.e. $x = 1$ and then the terms x_{12} and x_{13} have fractional values such that $0 < x_{12} < x_{13} < 1$.

Equations 6.2.6 and 6.2.7 are used as before in equation 6.2.3, in order to calculate the value of $\hat{G}^{-1}(\hat{q})$ at $\hat{q} = f(\hat{q}_1, \hat{q}_2)$ from the $G^{-1}(\overline{q})$ matrices previously calculated at each of the line interpolation points by

$$
\hat{G}^{-1}(\hat{q}) = \sum_{i=1}^{3or4} h_i(\overline{q})\ G_i^{-1}(\overline{q}) \qquad (6.2.8)
$$

7 Results

7.1 Introduction

Initially, the FEM models were generated by the FWR program and were then reproduced and checked against identical ABAQUS™ models. ABAQUS™ was then used to develop more complicated FEM models for use in the FWR method (e.g. spherical shell model). That is the global matrices (mass and stiffness) used in these examples were generated by the FWR program for the simple models and ABAQUS™ program for the larger more complicated models. The matrices were then reordered and partitioned according to the choices of "master" node points.

Calculations were preformed using both the Hermitian and Lagrangian interpolation methods in order to obtain a reduced solution for direct comparison to solutions of the full DOF model. The full DOF solution u_{full} is found from the equation

$$\mathbf{L}\, u_{full} = f \tag{7.1.1}$$

And the reduced solution $u_{Reduced}$ is calculated from the reduced DOF equation

$$\bar{\mathbf{L}}_0\ u_m = f_m \tag{7.1.2}$$

And the reconstructed "slave" displacement solution

$$u_s = \beta\, u_m \tag{7.1.3}$$

Hence, the total reduced solution ("master" and reconstructed "slave" results) is then

$$u_{Reduced} = \begin{pmatrix} u_m \\ u_s \end{pmatrix} \tag{7.1.4}$$

The Hermitian Interpolation method allows 2D rectangular window interpolation (4^{pt}) and 1D linear interpolation with 2 and 3 or 4^{fixed} or variable interior point location. Note that the Hermitian Interpolation method includes the function interpolation and derivative information, and for the 1D cases, with 3 or 4 points, interior window information. The Lagrangian Interpo-

lation method allows 2D rectangular window interpolation with 4, 5, 8 or 9 point two dimensional shape functions and 1D linear interpolation with 2 and 3 or 4 fixed or variable interior point location. The Lagrangian Interpolation method for 2D rectangular interpolation with 5, 8 or 9 point locations and 1D linear 3 or 4 points have interior window information.

The interpolation method for the 2D window was performed on several of the following examples. However, since the frequency sweeps were confined along the real frequency axis, the 2D analyses gave identical results when compared to the appropriate 1D analyses. The 2D Hermitian interpolation method has two interpolation (inversion) points on the real axis, hence its results were the same as the 1D 2^{pt} Hermitian interpolation method. Similarly, the 2D Lagrangian method which has three interpolation points on the real frequency axis was identical to the 1D 3^{pt} Lagrangian method.

In all of the following examples, results are shown for the 1D interpolation method confined to the real frequency axis. Also the analyses for all examples were performed using the higher point interpolations, 4^{pt} Hermitian and 4^{pt} Lagrangian. These methods gave more accurate results for comparison to the full DOF solutions.

7.2 Error Analysis

Error analysis was performed utilizing the relative error of the infinity norms of the matrices. This relative error is

$$\text{Relative Error} = \frac{\left\| \left(\hat{A} - A \right) \right\|_\infty}{\| A \|_\infty} \tag{7.2.1}$$

With the infinity norm of A is defined as $\| A \|_\infty = \max_{1 \le i \le m} \sum_{j=1}^{n} \left| a_{ij} \right|$ where A is an m x n matrix. The relative error as defined above is a measure of the decimal digit accuracy of $\hat{A}$ as compared to A.

From the definition of the relative error equation (7.2.1) and the resultant solutions for the full DOF problem (via 7.1.1) and the reduced DOF problem (via 7.1.2-7.1.4), the relative error is then written as

$$\text{Relative Error} = \frac{\left\| \left(u_{reduced} - u_{full} \right) \right\|_{\infty}}{\left\| u_{full} \right\|_{\infty}} \tag{7.2.2}$$

Where $u_{reduced}$ and u_{full} represent the entire solution. This relative error represents a stringent global error estimate.

The magnitude of the displacement for individual DOFs as obtained from the full DOF solution as well as from reduced solutions from the interpolation methods were directly compared. Percent error analysis was performed on specific DOFs to directly compare the solutions. The $\Delta\%$ or the percent error is defined as

$$\Delta\% = \left(\frac{\left| u_{reduced} - u_{full} \right|}{\left| u_{full} \right|} \right)_{DOF} * 100 \tag{7.2.3}$$

Where $u_{reduced}$ and u_{full} here represent the magnitude of the displacement solution of a specific individual DOF.

7.3 Eigenvalue Analysis

Eigenvalue analysis for all models was performed with ABAQUS™ and LAPACK™ routines for inclusion in the data analyses. The eigenvalue problem can be written in two ways, the standard eigenvalue form as

$$\mathbf{A}\, x = \lambda x \tag{7.3.1}$$

Or the generalized eigenvalue form as

$$\mathbf{A}\, x = \lambda^2 \mathbf{B} x \tag{7.3.2}$$

The frequency dependent mechanical system equation is

$$\left(-\omega^2 \mathbf{M} + i\omega \mathbf{D} + \mathbf{S}\right)\hat{u} = \hat{f} \tag{7.3.3}$$

Where **M**, **D** and **S** are the mass, damping and stiffness matrices respectively. The term $\hat{u}$ is the Fourier transform, $F = e^{i\omega t}$, of the system displacements and $\hat{f}$ is the Fourier transform of the system loads.

From equation (7.3.3), if the mechanical system is assumed to have *no* damping term present, i.e. $\left(-\omega^2 \mathbf{M} + \mathbf{S}\right)\hat{u} = \hat{f}$, then the generalized eigenvalue problem is written via equation (7.3.1) as

$$\mathbf{S}\, x = \lambda^2 \mathbf{M} x \tag{7.3.4}$$

Where only the mass and stiffness matrices are present.

This equation may be solved for the eigenvalues directly by use of the LAPACK™ routines LA_SYGV or LA_GEGV for symmetric or non-symmetric matrices respectively. Alternatively this equation may be solved by reformulation via the Cholesky decomposition of the mass matrix, **M**, as

$$\mathbf{M} = U^T U \tag{7.3.5}$$

The decomposed mass matrix is then used to transform the generalized eigenvalue problem (7.3.2) into the standard eigenvalue problem (7.3.1) as

$$\hat{\mathbf{A}}\, x = \nu x \tag{7.3.6}$$

Where the matrix $\hat{\mathbf{A}}$ is

$$\hat{\mathbf{A}} = \left(U^T\right)^{-1} \mathbf{S} U \tag{7.3.7}$$

And the eigenvalues are defined as $\nu \equiv \lambda^2 = (2\pi\omega)^2$. The eigenvalues may now be solved for by use of the LAPACK™ routines LA_POTRF, LA_SYGST and LA_SYEV.

The generalization of the standard eigenvalue problem to nonlinear eigenvalues is

$$\left(\mathbf{A}\lambda^2+\mathbf{B}\lambda+\mathbf{C}\right)\ x=\ 0 \tag{7.3.8}$$

Which can be seen to be a quadratic equation in λ. This nonlinear problem can be transformed to a 2n x 2n "standard" eigenvalue problem by

$$\begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{A}^{-1}\mathbf{C} & -\mathbf{A}^{-1}\mathbf{B} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} x \\ y \end{pmatrix} \tag{7.3.9}$$

Where **I** is the Identity matrix, the matrix $\hat{\mathbf{A}}$ is defined as

$$\hat{\mathbf{A}} \equiv \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{A}^{-1}\mathbf{C} & -\mathbf{A}^{-1}\mathbf{B} \end{pmatrix} \tag{7.3.10}$$

And

$$\bar{x} = \begin{pmatrix} x \\ y \end{pmatrix} \tag{7.3.11}$$

So the 2n x 2n "standard" eigenvalue problem now becomes

$$\hat{\mathbf{A}}\ \bar{x}\ =\ v\bar{x} \tag{7.3.12}$$

Applying this reformulation to the mechanical system equation (7.3.3) when damping *is* present, the matrix $\hat{\mathbf{A}}$ for the standard eigenvalue problem becomes

$$\hat{\mathbf{A}} \equiv \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{S} & -\mathbf{M}^{-1}\mathbf{D} \end{pmatrix} \tag{7.3.13}$$

And then the eigenvalues are now defined as

$$v\ \equiv\ i\lambda\ =\ 2\pi i\omega \tag{7.3.14}$$

NOTE: For this standard formulation it is assumed that it is possible to obtain the inverse of the mass matrix, i.e. $\mathbf{M}^{-1}$ exists.

When the mass matrix in *not* invertable, i.e. there are zero values along the diagonal of the matrix, it is possible to place small values for the zero terms (several orders of magnitude below the lowest value) on the diagonal. The presence of these small non-zero diagonal terms should now allow the inversion of the mass matrix directly, without significantly effecting the accuracy.

Alternatively in the event that the mass matrix is *not* invertable, the generalized nonlinear eigenvalue problem (7.3.8), i.e. when damping is present, can be transformed to a 2n x 2n "generalized" eigenvalue problem by

$$\begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{C} & -\mathbf{B} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{A} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{7.3.15}$$

Or

$$\begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{S} & -\mathbf{D} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{7.3.16}$$

The generalized eigenvalue problem is then becomes

$$\hat{\mathbf{A}}\ \bar{x} = \nu \hat{\mathbf{B}}\, \bar{x} \tag{7.3.17}$$

Where the terms are now $\hat{\mathbf{A}} \equiv \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{S} & -\mathbf{D} \end{pmatrix}$, $\hat{\mathbf{B}} \equiv \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{pmatrix}$, and $\nu \equiv i\lambda = 2\pi i\omega$.

NOTE: For both the standard and generalized eigenvalue problems defined above for the nonlinear case 2n eigenvalues are obtained which will result in $\mathbf{DET}\left|\hat{\mathbf{A}} - \lambda \mathbf{I}\right| = 0$ for the standard eigenvalue problem and $\mathbf{DET}\left|\hat{\mathbf{A}} - \lambda \hat{\mathbf{B}}\right| = 0$ for the generalized eigenvalue problem. These eigenvalues also will satisfy $\mathbf{DET}\left|\left(\lambda^2 \mathbf{A} + \lambda \mathbf{B} + \mathbf{C}\right)\right| = 0$.

Both of the above formulations for the nonlinear eigenvalue problem in the case where damping *is* present were tested via the LAPACK™ routines LA_GEEV and LA_GEGV. Both methods resulted in identical eigenvalues. These values were checked for consistency by applying the appropriate matrix determinant calculation.

7.4 Program Description

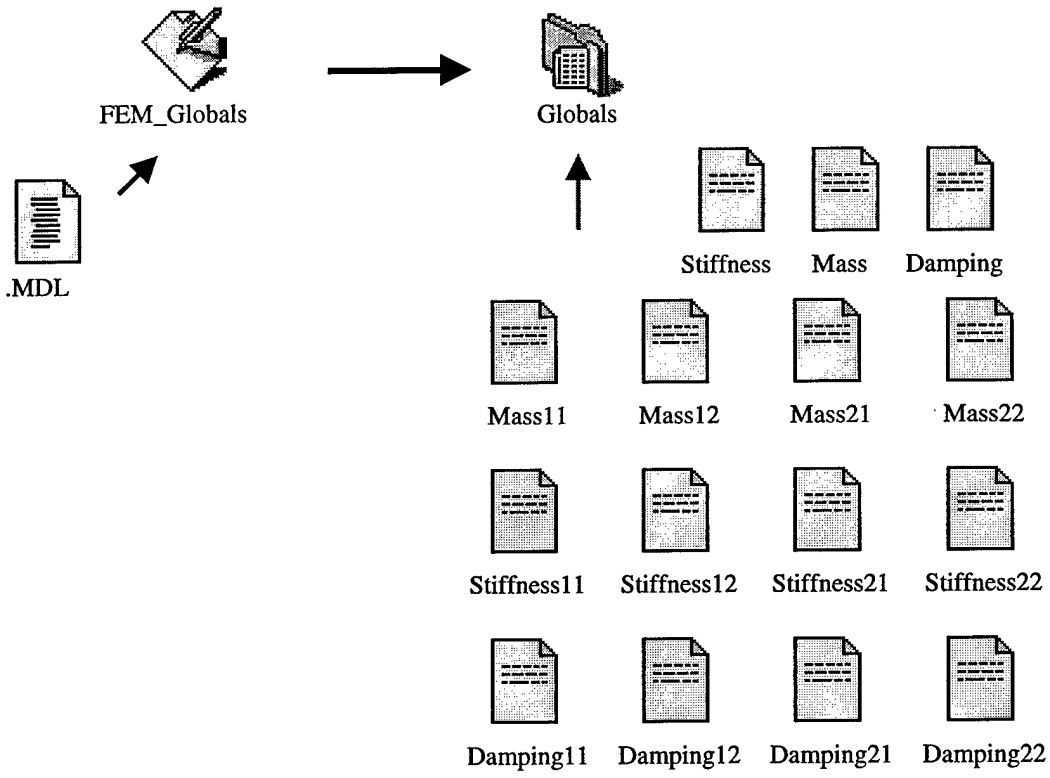
All computational codes were developed in-house at NRL using standard FORTRAN 90 (F90). The codes were initially written and tested on an Apple Macintosh™ Power PC computer using Apple Macintosh Programmers Workshop™ (MPW) and Absoft ProFortran™ version 6.0. These codes have been successfully ported to a Windows based PC platform. Fortran 90 was chosen because of its many programming features. The dynamic memory allocation of arrays, access to subarrays directly by indexing, usage of pointers as well as many implicit matrix routines are feature which make F90 very attractive for developing the necessary computational codes for this methodology. Additionally F90 offers the option of developing parallel codes for High Performance Computing (HPC) platforms. In F90, codes can easily be developed in modular form with the "modules" (.mod file) acting similarly to linked static libraries. This allows the development, testing and re-use of routines in multiple programs. Standard routines, such as LU decomposition, Cholesky decomposition, eigenvalue solvers, etc., were implemented from the LAPACK™ F90 version 2.0. LAPACK was incorporated in order to minimize the need to develop standard matrix computational routines, but yet allow broad cross-platform compatibility.

Brief descriptions of the major codes for the FWR method and a listing of the working directory and of program/routine names follow in this section.

Calculation of Global and Partitioned Matrices

FEM_Globals

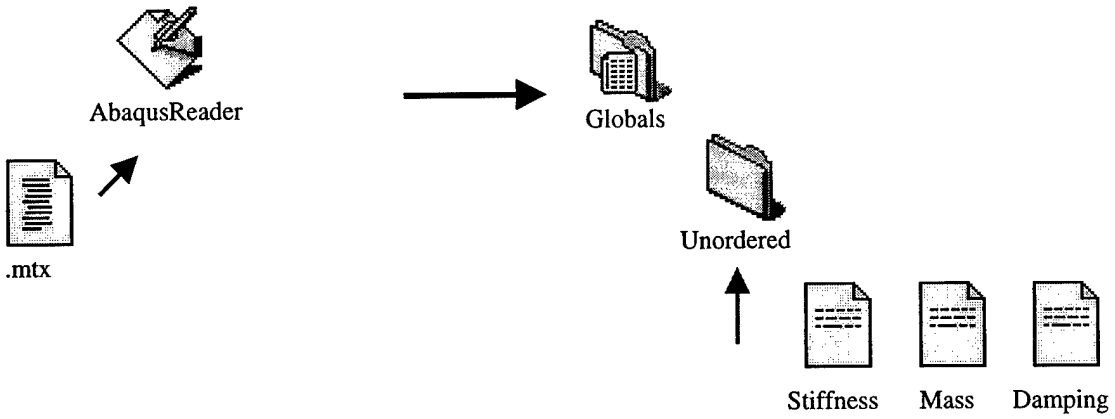
This is an in-house developed code, which reads a discretized model file and calculates the global mass and stiffness matrices for the FEM model. These full global matrices may either be generated as un-ordered, i.e. numbered as in the original FEM model, or as re-ordered according to a "master" node list. Note that for the re-ordered option that both the full global matrices and the partitioned matrices are saved in packed binary format. Also a zero damping matrix is output. Because of the limited element types able to be programmed, this program was initially used in order to test the subsequent programs for simple models.



AbaqusReader

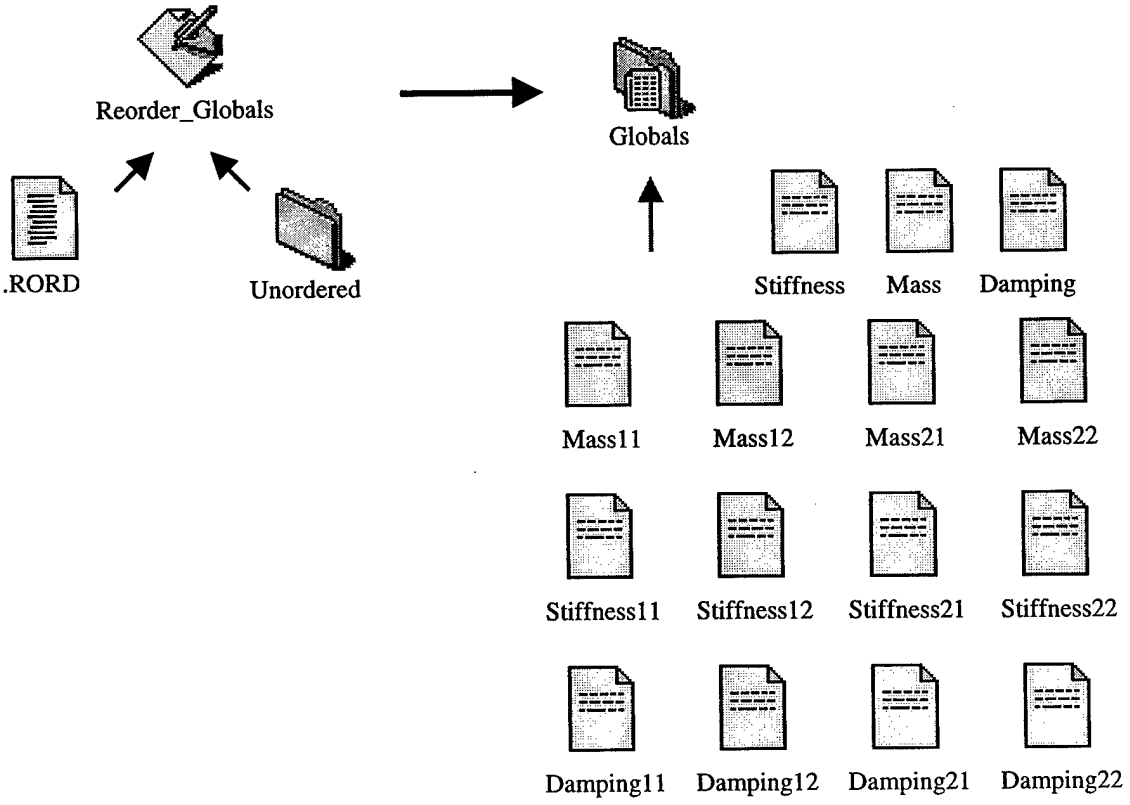
This program reads the output file from ABAQUS™, which contains the computed stiffness and mass matrices stored in lower triangular form. The program generates the full symmetry global stiffness and mass matrices and outputs them in packed binary format. Note that these

matrices are in the un-ordered form, i.e. numbered according to the FEM model. A zero damping matrix is also output.



Reorder_Globals

This program applies a permutation operation on the full global mass, stiffness, and damping matrices based on a "master" node list. The re-ordered full global matrices and the partitioned matrices are then output in packed binary format.



RayleighDamping

The Rayleigh damping matrix is calculated from the global mass and stiffness matrices by

$$\mathbf{D} \equiv \alpha\mathbf{M} + \beta\mathbf{S} \tag{7.4.1}$$

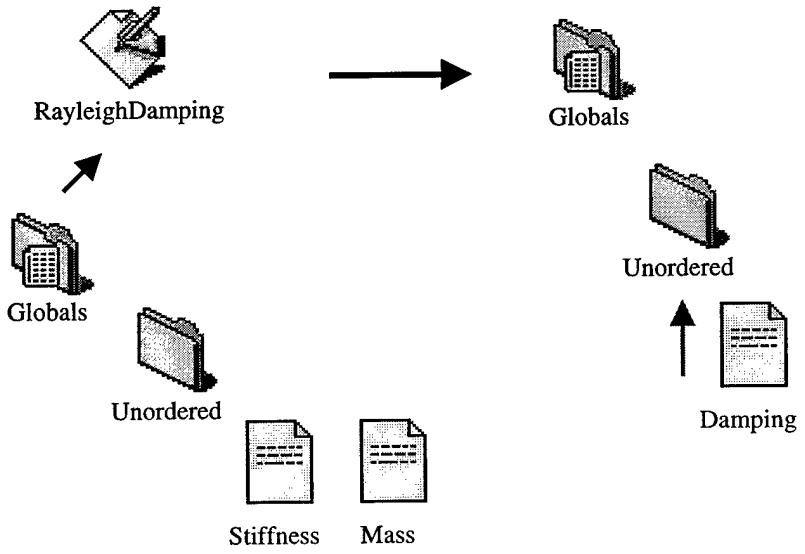
either from input Rayleigh damping coefficients α and β, or from two supplied resonant frequencies, ω_1 and ω_2, and their respective percentages of critical damping, ξ_1 and ξ_2. The Rayleigh damping coefficients are found as

$$\beta = \frac{2(\omega_2\xi_2 - \omega_1\xi_1)}{\omega_2^2 - \omega_1^2} \tag{7.4.2}$$

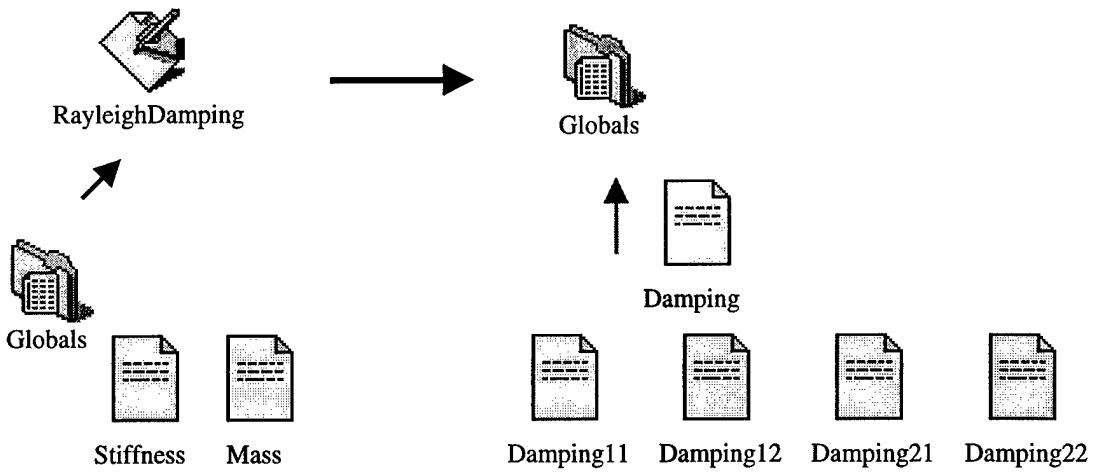
and

$$\alpha = 2\omega_1\xi_1 - \beta\omega_1^2 = \omega_1(2\xi_1 - \beta\omega_1). \tag{7.4.3}$$

Rayleigh damping may either be calculated for unordered *or* re-ordered matrices. If the calculation is for the unordered matrices, the above program, "Reorder_Globals" must be used to obtain the re-ordered full global matrices and the partitioned matrices for use in the FWR method programs.



OR


RayleighDamping
Globals
Damping
Globals
Stiffness
Mass
Damping11
Damping12
Damping21
Damping22

Calculation of Frequency Window Reduction Method

FWR_Constructors

From the input frequency window location and size, the interpolation parameters and the saved partitioned global matrices, the interpolation constructor matrices may be calculated for both Lagrangian and Hermitian interpolation methods. For the Lagrangian interpolation method (Section 6) at each interpolation point, $2 \leq N \leq 4$, the inverse of

$$G(q) = \tilde{\mathbf{L}}_{22}(q) - \alpha(q)\ \tilde{\mathbf{L}}_{12}(q) \tag{7.4.4.}$$

is calculated and saved in packed binary format.

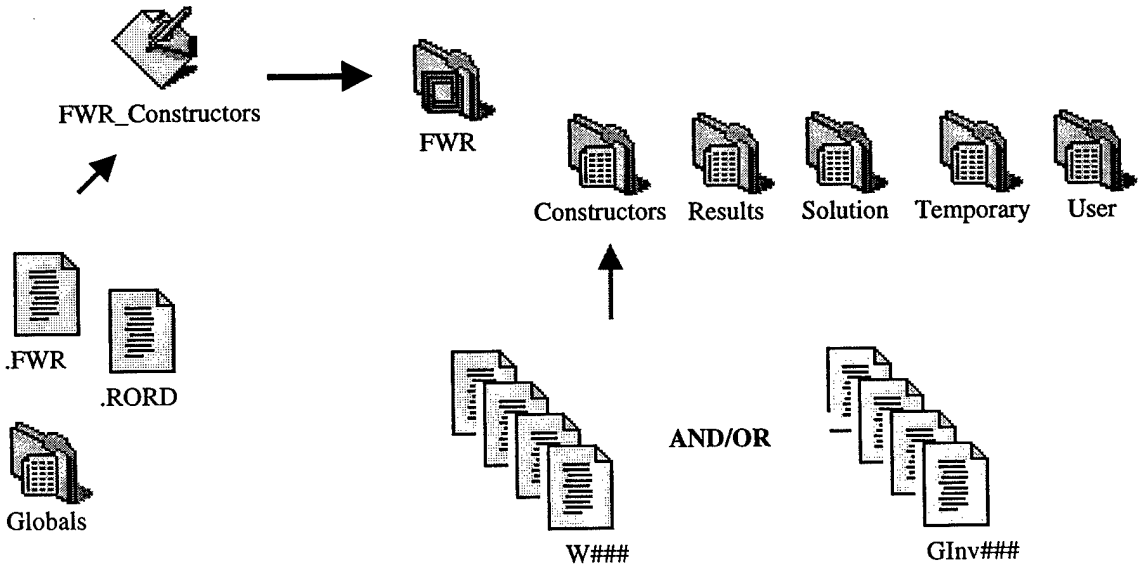
The interpolation constructor matrices, $W(\bar{q}_j)$, for the Hermitian interpolation method for a 1D linear interpolation with $2 \leq N \leq 4$ (Section 5) are

$$\begin{pmatrix} W_1 \\ W_2 \\ \vdots \\ W_{2N} \end{pmatrix} = B^{-1}(\bar{q}) \circ \begin{pmatrix} G^{-1}(\bar{q}_1) \\ \dfrac{\partial G^{-1}(\bar{q}_1)}{\partial q} \\ G^{-1}(\bar{q}_2) \\ \dfrac{\partial G^{-1}(\bar{q}_3)}{\partial q} \\ \vdots \\ G^{-1}(\bar{q}_N) \\ \dfrac{\partial G^{-1}(\bar{q}_N)}{\partial q} \end{pmatrix} \tag{7.4.5}$$

and

$$W(\bar{q}_j) = B^{-1}(\bar{q}_j) \circ \begin{pmatrix} G^{-1}(q_1,q_2) \\ \dfrac{\partial \hat{G}^{-1}(q_1,q_2)}{\partial q_1} \\ \dfrac{\partial \hat{G}^{-1}(q_1,q_2)}{\partial q_2} \\ \dfrac{\partial^2 \hat{G}^{-1}(q_1,q_2)}{\partial q_1 \partial q_2} \\ \vdots \\ \vdots \\ G^{-1}(q_1+w,q_2+h) \\ \dfrac{\partial \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_1} \\ \dfrac{\partial \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_2} \\ \dfrac{\partial^2 \hat{G}^{-1}(q_1+w,q_2+h)}{\partial q_1 \partial q_2} \end{pmatrix} \tag{7.4.6}$$

for the 2D rectangular window Hermitian interpolation. The $W(\bar{q}_j)$ matrices are saved a packed binary files for the next program calculation.



FWR_L0s

This program calculates the full DOF system and/or the reduced DOF system matrix for a given interpolated frequency window and a given frequency sweep within the window. Note that the appropriate constructor matrices must have been previously calculated by "FWR_Constructors".

Having found the Hermitian constructor matrices, $W(\bar{q}_j)$, for a 2D rectangular window at the points $\bar{q}_j$, $\hat{G}^{-1}(\hat{q})$ may be interpolated for a given value of the frequency in generalized coordinates $\hat{q} = f(\hat{q}_1,\hat{q}_2)$ within the window as

$$\hat{G}^{-1}(\hat{q}) \cong C_0(\hat{q}_1,\hat{q}_2)\ W(\bar{q}_j) \tag{7.4.7}$$

where

$$C_0(\hat{q}_1,\hat{q}_2) = \begin{pmatrix} 1 & \hat{q}_1 & \hat{q}_1^2 & \hat{q}_1^3 & \hat{q}_2 & \hat{q}_1\hat{q}_2 & \hat{q}_1^2\hat{q}_2 & \hat{q}_1^3\hat{q}_2 \\ \hat{q}_2^2 & \hat{q}_1\hat{q}_2^2 & \hat{q}_1^2\hat{q}_2^2 & \hat{q}_1^3\hat{q}_2^2 & \hat{q}_2^3 & \hat{q}_1\hat{q}_2^3 & \hat{q}_1^2\hat{q}_2^3 & \hat{q}_1^3\hat{q}_2^3 \end{pmatrix} \tag{7.4.8}$$

Alternatively, the $\hat{G}^{-1}(\hat{q})$ may be approximated at any point on a 1D line, $\hat{q} = f(\hat{q}_1,\hat{q}_2)$, as

$$\hat{G}^{-1}(\hat{q}) = W_1(\bar{q}) + \hat{q}\ W_2(\bar{q}) + \hat{q}^2\ W_3(\bar{q}) + \cdots + \hat{q}^{[(Order+1)N]-1}\ W_{(Order+1)N}(\bar{q}) \tag{7.4.9}$$

where the order of the Hermitian is restricted to first order and $2 \le N \le 4$.

The Lagrangian methods, having calculated the appropriate, $G^{-1}(\bar{q})$, the 2D interpolation is

$$\hat{G}^{-1}(\hat{q}) = \sum_{i=1}^{\#} h_i(\hat{r},\hat{s})\ G_i^{-1}(\bar{q}) \tag{7.4.10}$$

and the 1D linear interpolation is

$$\hat{G}^{-1}(\hat{q}) = \sum_{i=1}^{\#} h_i(\hat{r})\ G_i^{-1}(\bar{q}). \tag{7.4.11}$$

The number of interpolation points is an input parameter and the parametric coordinates are found from the frequency in generalized coordinates.

The having calculated $\hat{G}^{-1}(\overline{q})$, the matrix $\beta(\hat{q})$ is given as

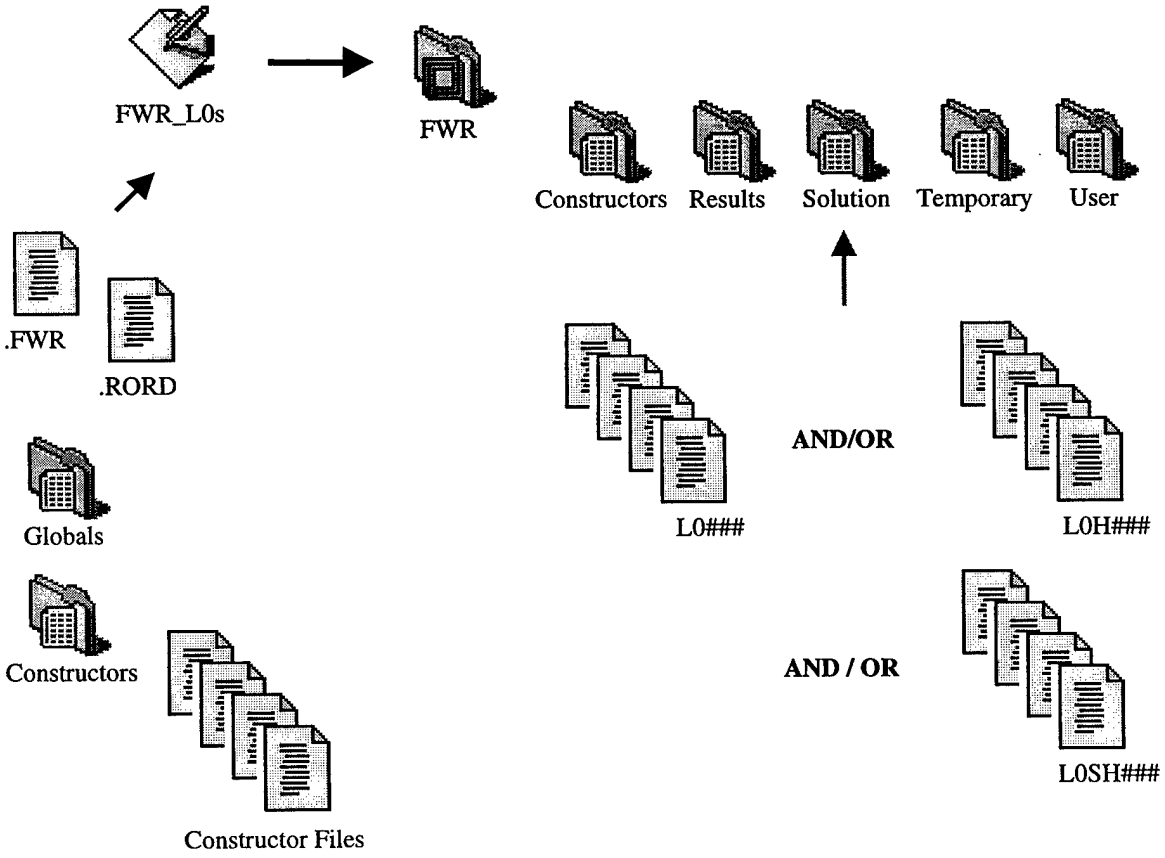
$$\beta(\hat{q}) \cong \hat{G}^{-1}(\hat{q})H(\hat{q}) \tag{7.4.12}$$

for both interpolation methods where $H(q) \equiv \alpha(q)\,\tilde{\mathbf{L}}_{11}(q) - \tilde{\mathbf{L}}_{21}(q)$. Finally the reduced DOF system constitutive matrix is

$$\overline{\mathbf{L}}_0(\hat{q}) = \tilde{\mathbf{L}}_{11}(\hat{q}) + \tilde{\mathbf{L}}_{12}(\hat{q})\ \beta(\hat{q}) \tag{7.4.13}$$

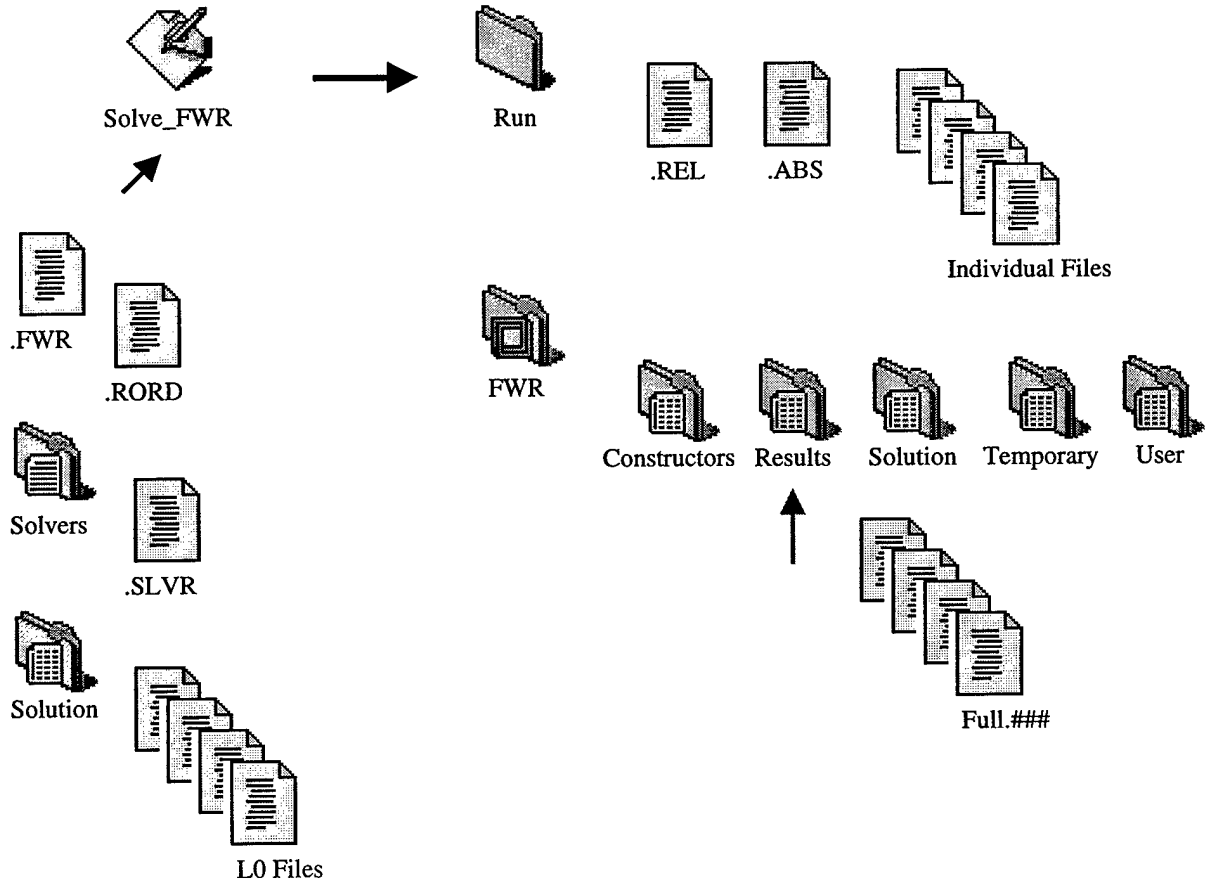
Both $\overline{\mathbf{L}}_0(\hat{q})$ and $\beta(\hat{q})$ are saved in one packed binary format file. Also the full DOF system matrix may be calculated and saved from,

$$\overline{\mathbf{L}}_0 \equiv q^2\mathbf{M} + q\mathbf{D} + \mathbf{S} \tag{7.4.14}$$



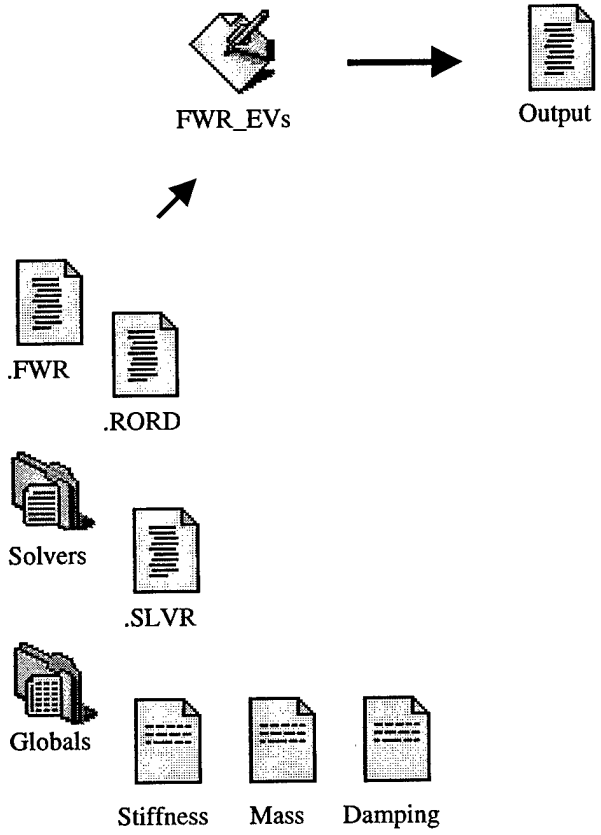
Solve_FWR

With the data calculated above, this program solves the full and reduced DOF system matrices from, $\overline{\mathbf{L}}_0$, which are saved as a function of frequency across an interpolation window. This program now solves for a given load and boundary conditions the equation $\overline{\mathbf{L}}_0\,\hat{u} = \hat{f}$ for the full DOF system and/or the equation $\overline{\mathbf{L}}_0\,\hat{u}_m = \hat{f}_m$ for the reduced DOF system. Also for the reduced system, the total reconstructed solution is calculated from $u_s = \beta\, u_m$. The solutions as a function of frequency, individual DOF solutions as well as the error analysis described in section 7.2 are all outputs of this program.



FWR_EVs

This program calculates the constrained *or* unconstrained eigenvalues of the full DOF model according to the procedure described in section 7.3 and LAPACK routines [12]. Note that the program will calculate the eigenvalues for systems with and without damping.



Application Directory and Working Directories

Globals — Files of full and partitioned Mass, Stiffness & Damping Matrices

Unordered — Files of full Unordered Global Matrices

Models — Contains "NameList" text files of the FEM Model Files

Elements — Contains "NameList" text files of the FEM Element Types

Solvers — Contains "NameList" text files of the Solver Files

Modules

Drivers

Mains — Module, Driver and Main Source Text Files

Objs — Contains Compiled Object Files, ".o" and ".mod"

FWR — Storage Directories for Frequency Window Reduction Calculations

Constructors

Results

Solution

Temporary

User

LAPACKMods

LAPACKLibs — LAPACK Modules and Pre-Compiled Libraries

Run — Directory for Storage of Frequency Window Reduction Output Files

Listing of Program and Routine Names

MAINS:

PROGRAM FEM_Globals
PROGRAM AbaqusReader
PROGRAM Reorder_Globals
PROGRAM RayleighDamping
PROGRAM FWR_Contructors
PROGRAM FWR_L0s
PROGRAM Solve_FWR
PROGRAM FWR_EVs

DRIVERS:

SUBROUTINE Do_Global_Matrices
SUBROUTINE Do_Reorder_Global
SUBROUTINE Do_FWR_Contructors
SUBROUTINE Do_FWR_L
SUBROUTINE Do_Solve_FWR
SUBROUTINE Do_FWR_EV

MODULES:

MODULE nrtype
MODULE FEM_BOUNDS
MODULE FEM_CONTROL
MODULE FEM_INC
MODULE FEM_INC_FIXED

MODULE standard_file_names
    FUNCTION FILE_NAME(IO,ID)

MODULE sparseutil
    FUNCTION CHECK_FILE(ID,IO_NAME)
    FUNCTION CHECK_SIZES(M,N,MP,NP)
    FUNCTION CHECK_MATMUL(M,N,message)
    SUBROUTINE CHECK_SPARSE(IO_NAME,len,m,n)
    SUBROUTINE CHECK_THRESHHOLD(A,threshhold,OPTION)
    FUNCTION arth(first,increment,n)
    SUBROUTINE GET_SPARSE(IO_NAME,spa)
    SUBROUTINE PUT_SPARSE(IO_NAME,spa)
    SUBROUTINE sparsein(a,threshhold,spa)
    SUBROUTINE sparseout(spa,threshhold,a)
    FUNCTION matrixout(spa,m,n)

```
        SUBROUTINE PUT_MATRIX(IO,ID,MP,NP,matrix)
        FUNCTION GET_MATRIX(IO,ID,MP,NP)

MODULE standard_interface
        SUBROUTINE GET_DATE_TIME(ID)
        FUNCTION CHECK_GLOBALS(N,OPTION)
        SUBROUTINE CHECK_GLOBALS_MESSAGE(TESTFLAG,OPTION)
        SUBROUTINE PRINT_GLOBALS_MESSAGE(IO,ID)
        FUNCTION OVERWRITE_MESSAGE(IOPT)
        SUBROUTINE
            CHECK_CONSTRUCTORS(w,h,HNUM,SFNUM,HExist,SFExist)
        SUBROUTINE
            CHECK_CONSTRUCTORS_MESSAGE(TESTFLAG1,TESTFLAG2)
        SUBROUTINE CONSTRUCTORS_OPTION(ISTART,IEND)
        SUBROUTINE
            FWR_L_OPTION(TESTFLAG1,TESTFLAG2,ISTART,IEND)
        SUBROUTINE GET_FEM_FILE(ID,IO_NAME)
        SUBROUTINE
           CHECK_THRESHHOLD_OPTION_d(MP,NP,matrix,MATRIX_NAME)
        SUBROUTINE
           CHECK_THRESHHOLD_OPTION_z(MP,NP,matrix,MATRIX_NAME)
        SUBROUTINE WRITE_MATRIX(X,MP,NP,M,N)
        SUBROUTINE
            PRINT_MATRIX_OPTION_d(MP,NP,matrix,MATRIX_NAME)
        SUBROUTINE
            PRINT_MATRIX_OPTION_z(MP,NP,matrix,MATRIX_NAME)
        FUNCTION TRACE_EXECUTION(IOPT)

    MODULE MTRXUTIL
        FUNCTION FACTORIAL(N)
        FUNCTION LA_LU_DETERMINANT(n,a)
        FUNCTION LA_LU_INVERSE(n,a)
        FUNCTION LA_LU_SOLVE(n,a,b,ID)
        FUNCTION IS_MATRIX_SQUARE(A)
        FUNCTION SYMMETRY_CHECK(A,threshhold)
        SUBROUTINE SORTER(N,A,B,C,ID)
        FUNCTION MATRIX_SYMMETRY(N,A)
        FUNCTION IDENTITY_MATRIX(N,X)
        FUNCTION GET_MATRIX_DIAGONAL(A)
        SUBROUTINE PUT_MATRIX_DIAGONAL(A,X,THRESHHOLD)
        FUNCTION ROW_COLUMN_ZERO_CHECK(N,A,ID)
        SUBROUTINE QUERY_MATRIX(A,OPTION)
        FUNCTION MATRIX_POWER(K,N,X)
        FUNCTION A_MATRIX_ms(m,n,mass,stiffness)
        FUNCTION A_MATRIX_msd(m,n,mass,stiffness,damping)
        SUBROUTINE EIGEN_VALUES_s(m,stiffness,IsSSym)
```

```
      SUBROUTINE EIGEN_VALUES_ms(m,mass,stiffness,IsMSym,IsSSym)
      SUBROUTINE EIGEN_VALUES_msd(m,mass,stiffness,damping, &
                  IsMSym,IsSSym,IsDSym)
      SUBROUTINE WRITE_EIGEN_VALUES(ID,K,m,WR,WI,BETA,IO)

MODULE FWRUTIL
      FUNCTION Complex_Magnitude(x,a)
      SUBROUTINE CHECK_Q_BOUNDS(IO,wnd,pnt,InWnd)
      FUNCTION GET_POINT_2D(ID,x0,w,h)
      FUNCTION GET_POINT_1D(ID,x0,w,h,x12,x13)
      FUNCTION dxi(ID,x,x0,w,h)
      FUNCTION L(ID,MP,NP,x,derivative,option)
      FUNCTION G(ID,MP,NP,x,derivative,option)
      FUNCTION H(ID,MP,NP,x,option)
      FUNCTION C(N,NUM,x1,x2)
      FUNCTION B(N,NUM,x0,w,h,x12,x13)
      FUNCTION C0(N,NUM,x1,x2)
      SUBROUTINE
           U_MATRICES(ID,MP,NP,NUM,x0,w,h,x12,x13,length,m,n,option)
      SUBROUTINE W_MATRICES(ID,MP,NP,NUM,x0,w,h,x12,x13,option)
      SUBROUTINE G_INVERESES(ID,MP,NP,NUM,x0,w,h,x12,x13,option)
      FUNCTION APPROX_G_INVERSE(ID,IO,NUM,MP,NP,x,x0,w,h,x12,x13)
      FUNCTION INTERPOLATION_FUNCTIONS_2D(M,MP,x,x0,w,h)
      FUNCTION
           INTERPOLATION_FUNCTIONS_1D(M,MP,x,x0,w,h,x12,x13)
      FUNCTION
           BETA_MATRIX(ID,IO,NUM,MP,NP,x,x0,wx,hx,x12,x13,option)
      FUNCTION  L0_MATRIX(ID,IO,NUM,MP,NP,x,x0,w,h,x12,x13
           ,BETAMATRIX,option)
      FUNCTION REDUCED_FWR_MATRICES(IO,MP,NP,BETAMATRIX)

MODULE USERUTIL
      FUNCTION ALFA(ID,MP,NP)
      FUNCTION ALPHA(ID,MP,NP,q)
      FUNCTION dALPHA(ID,MP,NP,q)
      FUNCTION ddALPHA(ID,MP,NP,q)
      FUNCTION DIRICHLET_FUNC(T)
      FUNCTION LOAD_FUNC(N,T)

MODULE GLOBALUTIL
      FUNCTION ELASTIC_MATRIX(ID,NP,MP,IPRINT,PROPERTY)
      SUBROUTINE INTEGRATION_PTS(IORDER,QUAD,RPT,SPT,TPT,
           RWT,SWT,TWT)
      FUNCTION LOCAL_COORD(N,MP,NP,IPRINT,XYZ,CONNECT)
      SUBROUTINE JACOBIAN(ID,M,N,IPRINT,XYZL,DSF,XJINV,DET)
      FUNCTION
```

```
        B_MATRIX(ID,M,N,MP,IDCMTRX,IPRINT,XYZL,SF,DSF,XJINV)
    FUNCTION H_MATRIX(ID,M,N,IDCMTRX,IPRINT,SF)
    SUBROUTINE LOCAL_to_GLOBAL(NUMDOF,M,N,LP,NP,CONNECT,
        local,global,LOOKUP,OFFSET)
    SUBROUTINE REORDER_GLOBAL(NDOF,N,NP,LOOKUP,
        global,reorder,NNDOF,ORGNLOFFSET,REORDOFFSET)
    SUBROUTINE SHAPE_FUNCTIONS(ID,M,N,R,S,T,SF,DSF)
    SUBROUTINE BAR_BEAM(M,N,MP,NP,R,S,T,SF,DSF)
    SUBROUTINE RECTANGULAR(M,N,MP,NP,R,S,T,SF,DSF)
    SUBROUTINE TRIANGULAR(M,N,MP,NP,R,S,T,SF,DSF)
    SUBROUTINE
        GLOBALS_OUTPUT(IO,NP,MP,MTRXTHSD,global,RENUMBER)
    SUBROUTINE GLOBAL_SUMMARY(IO,ID)
    FUNCTION OFF_SET_LIST(N,DOF_PER_NODE,LOOKUP)
    SUBROUTINE MASTER_NODE_LIST(N,M,NNMF,NNFM)
    FUNCTION QUAD_CHECK(IDELMT)
    FUNCTION GET_SIDE_NUM(IDELMT,NUMNEL)
    SUBROUTINE  READ_ELEMENT_TYPE(ID,EL_INFO,STF_INT,
          MSS_INT,DMP_INT,FRC_INT)
    SUBROUTINE PRINT_ELEMENT_TYPE(EL_INFO,STF_INT,
          MSS_INT,DMP_INT,FRC_INT)
    SUBROUTINE SET_ELEMENT_SIDE(EL_INFO)


MODULE GLOBALS
    SUBROUTINE GLOBAL_MATRICES
    SUBROUTINE STIFFNESS(N0,NP,MP,global,extra,IDLIST)
    SUBROUTINE MASS(N0,NP,MP,global,extra,IDLIST)
    SUBROUTINE DAMPING(N0,NP,MP,global,extra,IDLIST)

MODULE MODELUTIL
    SUBROUTINE ID_FEM_FILE(IO_NAME)
    SUBROUTINE CLEAR_ALLOCATED_MATRICES(I)
    SUBROUTINE FEMMODEL_FILE_IO(IO,IO_NAME)
    SUBROUTINE FEMMODEL_SUMMARY(IO)
    SUBROUTINE FWRMODEL_FILE_IO(IO,IO_NAME)
    SUBROUTINE FWRMODEL_SUMMARY(IO)
    SUBROUTINE CMRMODEL_FILE_IO(IO,IO_NAME)
    SUBROUTINE CMRMODEL_SUMMARY(IO)
    SUBROUTINE FEMSOLVER_FILE_IO(IO,IO_NAME)
    SUBROUTINE FEMSOLVER_SUMMARY(IO)
    SUBROUTINE REORDER_FILE_IO(IO,IO_NAME)
    SUBROUTINE REORDER_SUMMARY(IO)

MODULE SOLVERUTIL
    FUNCTION BCMATRIX_d(MP,NP,matrix)
```

```
FUNCTION BCMATRIX_z(MP,NP,matrix)
FUNCTION RHSFORCES(NP,T)
FUNCTION RHSBCS(NP,T,FORCES)
SUBROUTINE RUNGEKUTTA(MP,LP,NP,A1,B1,AINV,
        NODE_NUM,NODE_DOF)
SUBROUTINE CENTRALDIFFERENCE(MP,LP,NP,A1,B1,AINV,
        NODE_NUM,NODE_DOF)
SUBROUTINE NEWMARKSMETHOD(MP,LP,NP,A1,B1,AINV,
        PRETERMS,NODE_NUM,NODE_DOF)
SUBROUTINE RECONSTRUCTION(MP,LP,NP,F,U,V)
SUBROUTINE RUN_SOLVERS(NODE_NUM,NODE_DOF)
SUBROUTINE SOLVER_PREP(MP,A1,B1,AINV,PRETERMS)
SUBROUTINE EXTRACT_TUV(NODE_NUM,NODE_DOF)
SUBROUTINE WRITE_ERROR_HEADER(I,ID,IO,NORM)
SUBROUTINE WRITE_RESULTS(ID,IO,LP,NDOF,LHS,norm_id)
SUBROUTINE WRITE_ERRORS(ID,LP,NDOF,ID2,norm_id)
FUNCTION CALCULATE_ERRORS(I)
SUBROUTINE WRITE_ERROR_NORMS(ID,LP,ID2,norm_id)
```

8 Examples

8.1 3-Layer Steel-Polystyrene 1D Model

A simple one-dimensional (1D), 3-layer laminate was considered as a first test of the FWR method. The laminate was comprised of layers of steel and polystyrene as shown in Figure 8.1.1 with the properties listed in Table 8.1.1. The FEM model was formed from 26 ABAQUS™ 2D plain stress elements consisting of 46 node points representing 92 DOFs (2 degrees of freedom per node point). Fourteen "master" nodes (28 DOFs) were selected and are shown in the figure as "•" symbols. These were chosen as master nodes as they have applied loads and fixed boundary conditions in the FEM model. The boundaries between element groups of differing materials and element grouping midpoints were also chosen as master nodes. This represents a 70% reduction in the degrees of freedom, i.e. 92 DOFs for the full problem to 28 DOFs for the reduced problem. For the level of discretization, only frequencies up to 350 KHz are appropriate (assuming 8-10 elements are necessary to model the highest frequency).

Table 8.1.1 - Properties for Steel-Polystyrene Laminate

	Density	Young's Modulus	Poisson Ratio	Wave Velocity
Steel	7900 Kg/m^3	256.7 GPa	0.25	5700 m/sec
Polystyrene	1050 Kg/m^3	9.387 GP a	0.25	2990 m/sec

The largest valid frequency window of 0 to 350 KHz was examined with a 70% reduction in the degrees of freedom. Figure 8.1.2 shows the relative error for a frequency sweep over the entire frequency window of 0 to 350 KHz. The figure compares several different interpolation functions. It should be noted that at the end points and the two interior points (approximately 116.7 KHz and 233.3 KHz) the error is very small. This is because these points represent points at which the direct inversion for the solution is obtained; hence the solution is nearly exact. It can be seen that generally the 4^{pt} Lagrangian, because of the interior point information is better than the 2^{pt} Hermitian, even though the Hermitian has derivative information at the interpolation (end) points. Clearly the 4^{pt} Hermitian, with both interior point function and derivative information has

the lowest relative error as well as capturing more accurately several of the eigenvalues of the system, Table 8.1.2. The eigenvalues are represented on the plots as gray vertical lines. Not all the eigenvalues are captured because of the frequency sweep step size of 1 KHz is not fine enough to approach the singular eigenvalue.

Table 8.1.2 - Calculated Constrained Eigenvalues

Frequency KHz
0.909
7.666
19.522
23.211
61.726
71.666
119.722
143.067
167.134
182.101
256.836
266.268
300.511
312.869
318.306
407.034

The actual % error of the magnitude of the resultant displacement for the "master" DOF u_{1x} is show in Figure 8.1.3. The 4^{pt} Lagrangian and the 2^{pt} Hermitian interpolation functions show very large errors as compared to the full DOF solution. However, the 4^{pt} Hermitian method has generally less than 1% error over the entire frequency range. The comparison of the 4^{pt} Lagrangian and the 4^{pt} Hermitian interpolation to the full DOF solution for the magnitude of the displacement for the "master" DOF u_{1x} is show in Figure 8.1.4. It can be seen that that both produce reasonably good agreement. The % error of the magnitude of the "reconstructed" displacement for the "slave" DOF u_{15x} shown in Figure 8.1.5, as expected, is identical to the % error for "master" DOFs since the "slave" DOFs are reconstructed directly from the "master" DOF solution according to equation 2.14.

Using the frequency window range of 0 to 350 KHz for the interpolation functions and only performing a partial sweep of the frequency window, it is possible to see more clearly the effects the different interpolation functions have near an eigenvalue. The relative error for an interpolation window of 0 to 350 KHz but for a frequency sweep of 250 to 280 KHz with a step

size of 150 Hz is shown in Figure 8.1.6. It is easily seen that the 4^{pt} Lagrangian interpolation undershoots the eigenvalue where as the 2^{pt} Hermitian overshoots the eigenvalue and the 4^{pt} Hermitian more closely matches the eigenvalue response. This is more easily demonstrated in Figure 8.1.7 examining the magnitude of the displacement for the "master" DOF u_{1x}which shows how closely the respective solutions approach the full solution near this eigenvalue. These figures indicate how accurate or smooth the interpolation functions are within a given interpolation window.

In order to improve the accuracy of the FWR method, the frequency window interpolation range was reduced. This reduction in the window range allows for improvement of the interpolation function as well as a finer frequency sweep step size. Also with a reduced frequency window range, there are fewer eigenvalues within the window, which improves the overall accuracy. Figure 8.1.8 shows the relative error for the frequency window of 0 to 100 KHz where the sweep step size was 500 Hz. Dramatic improvements (several orders of magnitude) are seen in all the interpolation methods. This dramatic improvement is reflected in the % error for the magnitude of the displacement for the "master" DOF u_{1x} shown in Figure 8.1.9. Both 4^{pt} Lagrangian and the 2^{pt} Hermitian interpolation methods now show less than 1% error and the 4^{pt} Hermitian interpolation method is less that 0.00001 % error. The plot of the actual the magnitude of the displacement for the "master" DOF u_{1x} in Figure 8.1.10 shows virtually no distinction between the interpolated reduced solutions and the full DOF solution.

The further reduction of the interpolation frequency window to the range of 0 to 45 KHz with a frequency sweep step size of 150 Hz is shown in Figure 8.1.11. Again dramatic improvements (several orders of magnitude) are seen in all the interpolation methods and it is also seen in the % error of the magnitude of the displacement for the "master" DOF u_{1x} shown in Figure 8.1.12. Here the 4^{pt} Lagrangian and the 2^{pt} Hermitian interpolation function are less than 0.1% error with the 4^{pt} Hermitian interpolation method is less than 0.0000001 % error. Clearly for this frequency window range for all the interpolation functions, the magnitude of the displacement for the "master" DOF u_{1x} is numerically identical to the full DOF solution, Figure 8.1.13.

For the interpolation frequency window range of 0 to 45 KHz, the number of DOFs was dramatically reduced from 92 DOFs to 8 DOFs representing only the loaded and constrained node points. This represents a 91% DOF reduction. Comparing the relative error for only the 4^{pt} Lagrangian and the 4^{pt} Hermitian interpolation methods, it is seen in Figure 8.1.14 that despite such a drastic reduction in the number of degrees of freedom, very accurate results are obtained for this model. Even with a 91% DOF reduction the 4^{pt} Lagrangian interpolation shows approximately less than 1% error and the 4^{pt} Hermitian interpolation method is less that 0.001 % error in Figure 8.1.15. Also the actual magnitudes of the displacement for these interpolation methods as compared to the full DOF solution are nearly identical as shown in Figure 8.1.16.

3-Layer Steel-Polystyrene 1D Figures

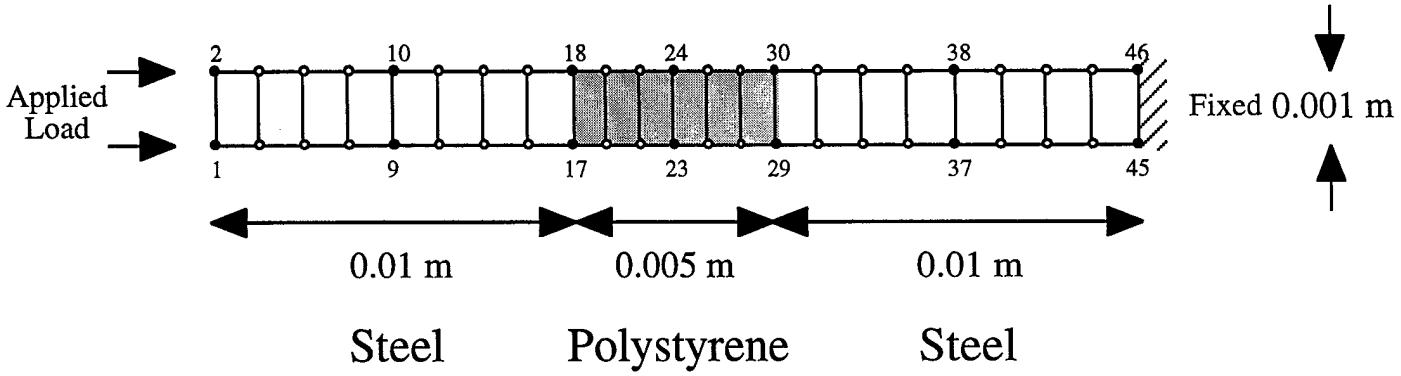


Figure 8.1.1 – 3-Layer Steel-Polystyrene Laminate 1D Model

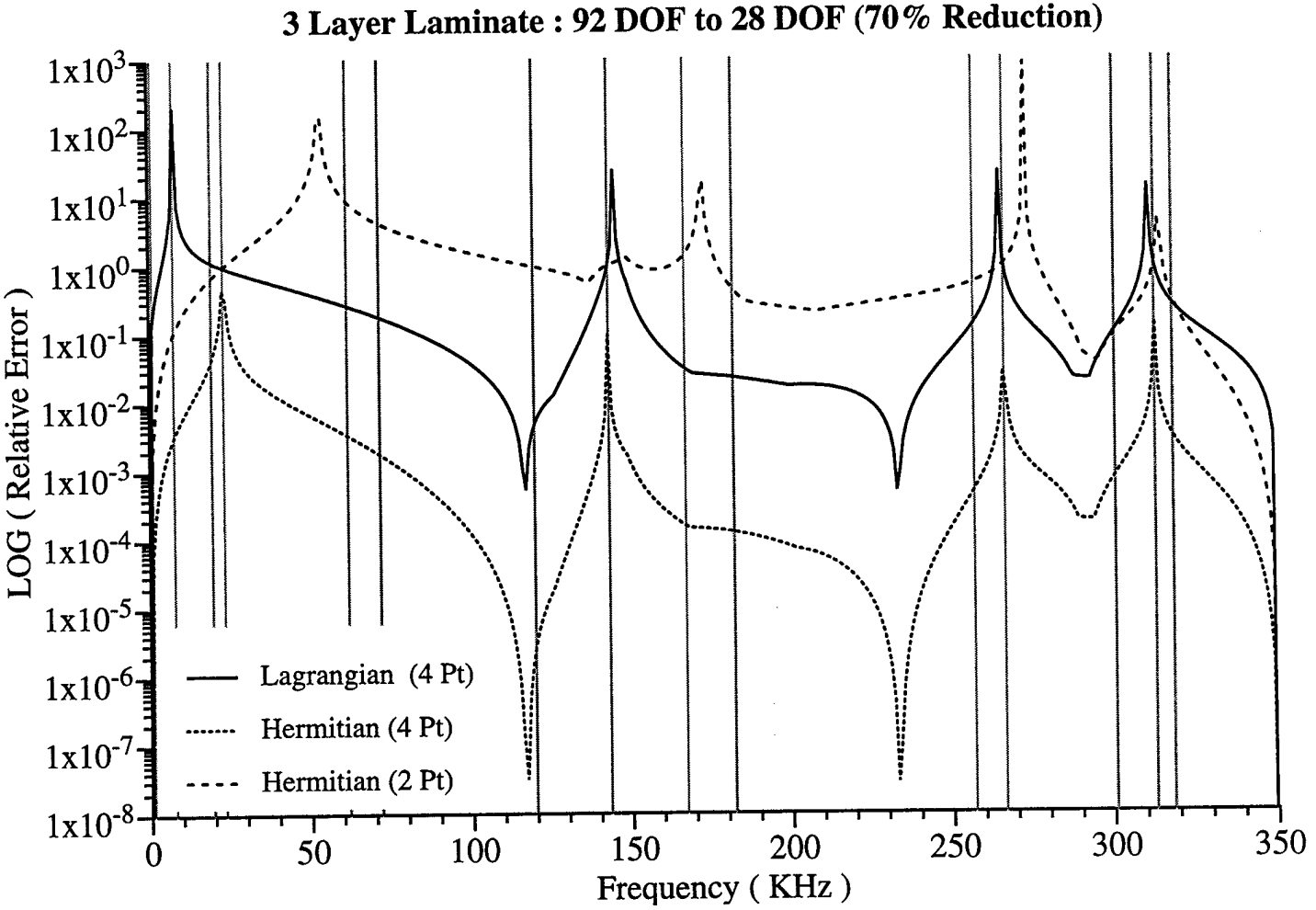


Figure 8.1.2 – Relative Error comparing interpolation functions for the frequency range of 0 to 350 KHz and a frequency step size of 1 KHz.

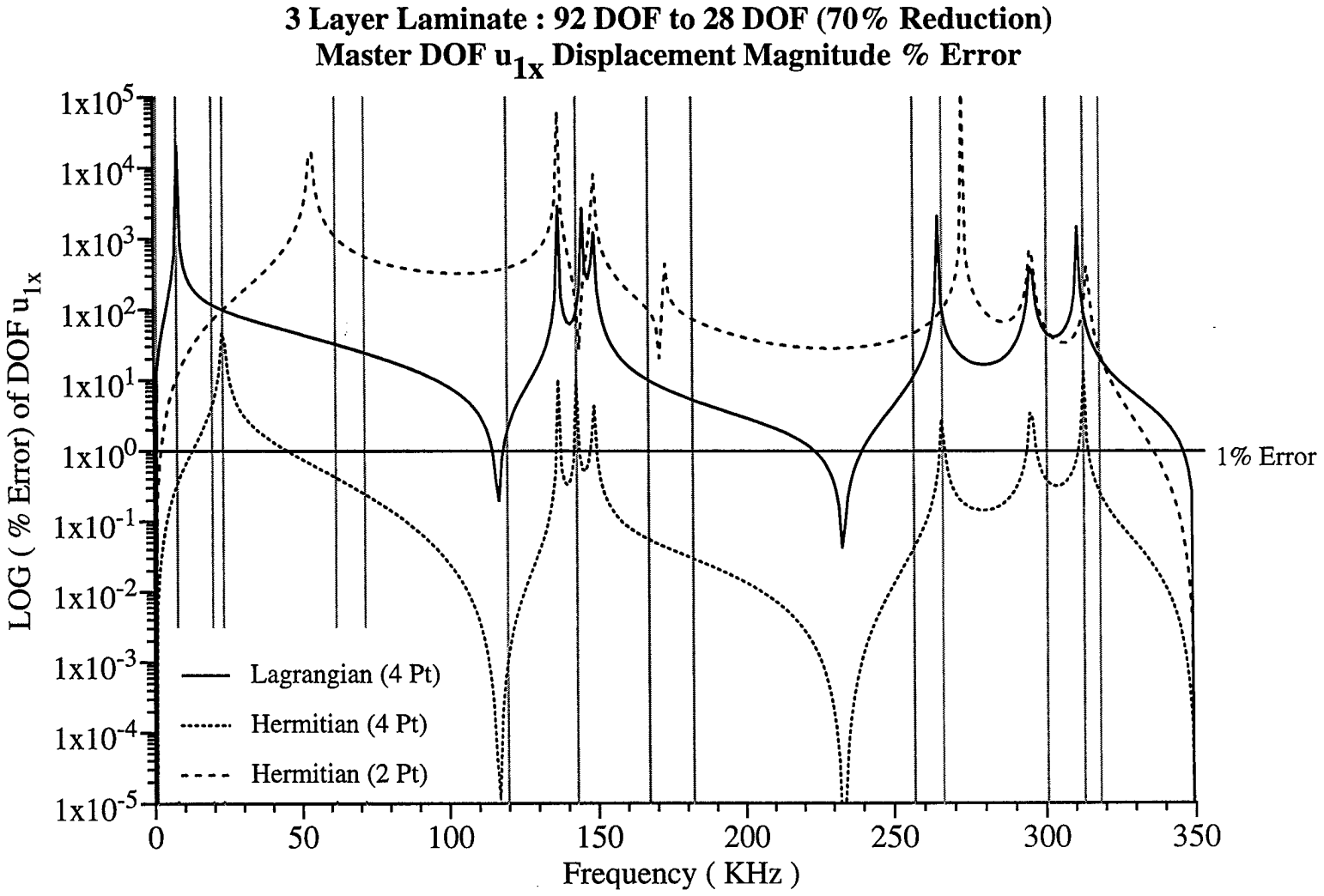


Figure 8.1.3 – % Error for the "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing interpolation functions for the frequency range of 0 to 350 KHz and a frequency step size of 1 KHz.

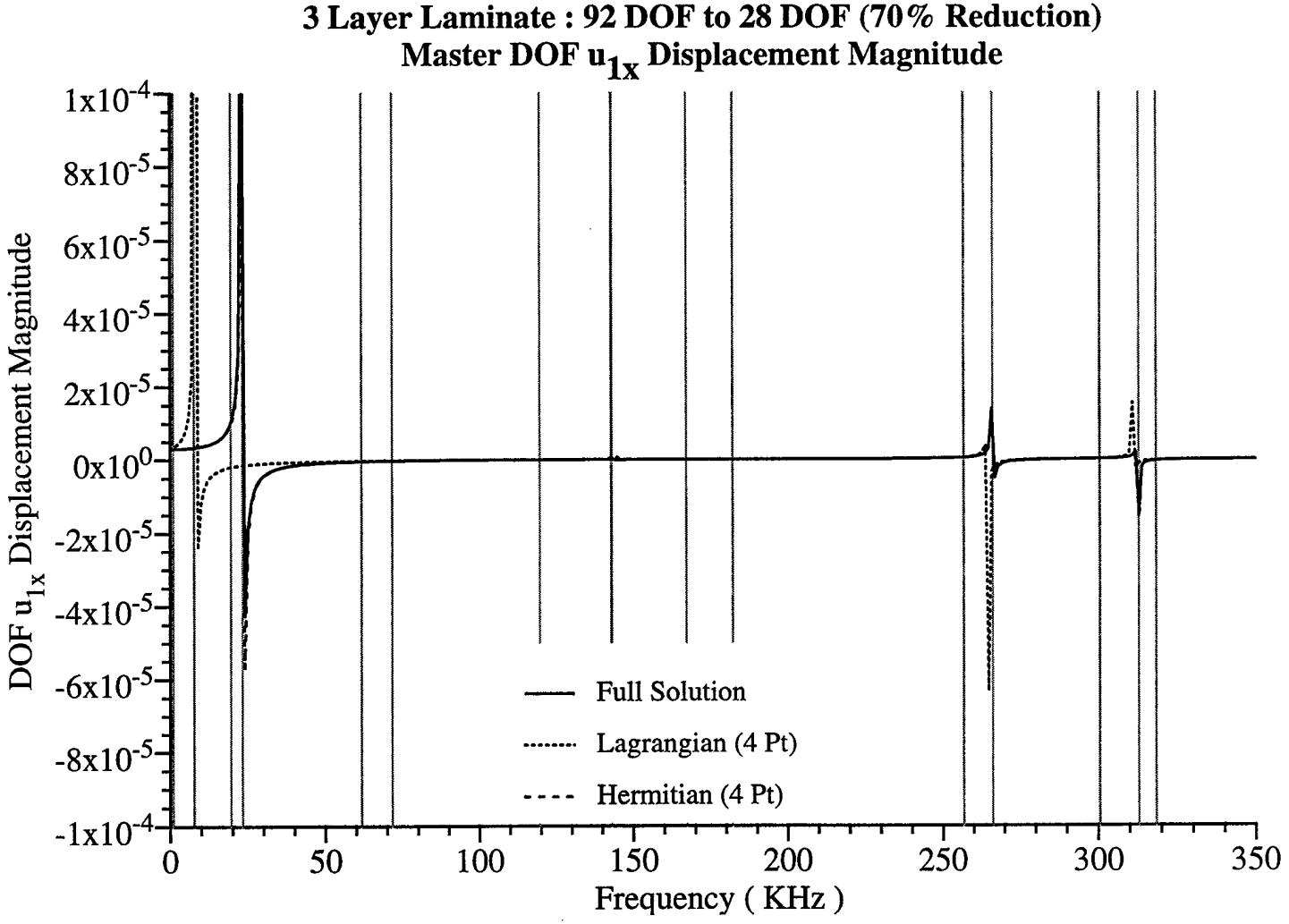


Figure 8.1.4 – The "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing the interpolation functions, 4pt Lagrangian and Hermitian, to the full DOF solution for the frequency range of 0 to 350 KHz.

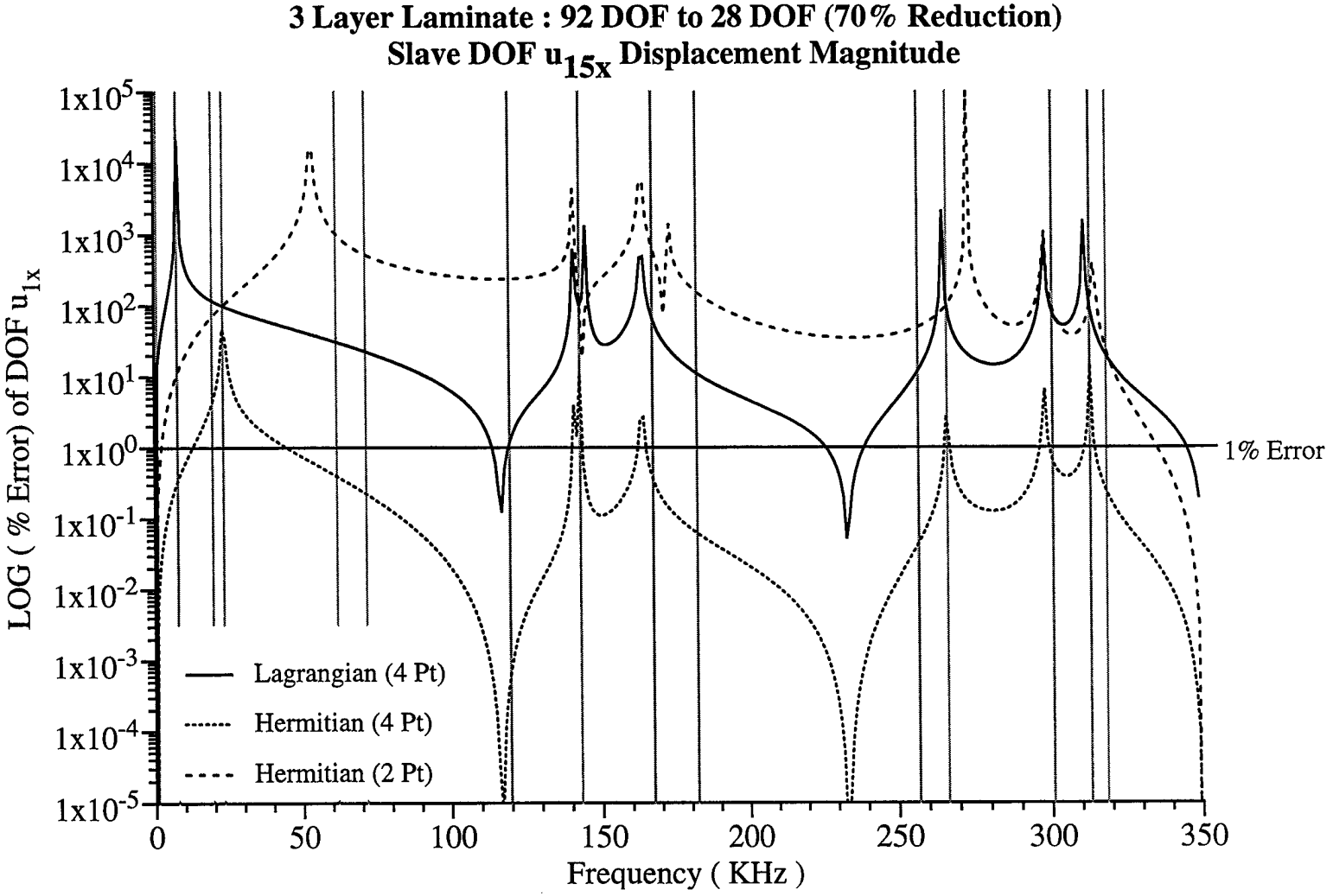


Figure 8.1.5 – The "slave" DOF $|u|_{15x}$ (magnitude of the displacement) comparing two interpolation functions to the full DOF solution for a frequency range of 0 to 350 KHz.

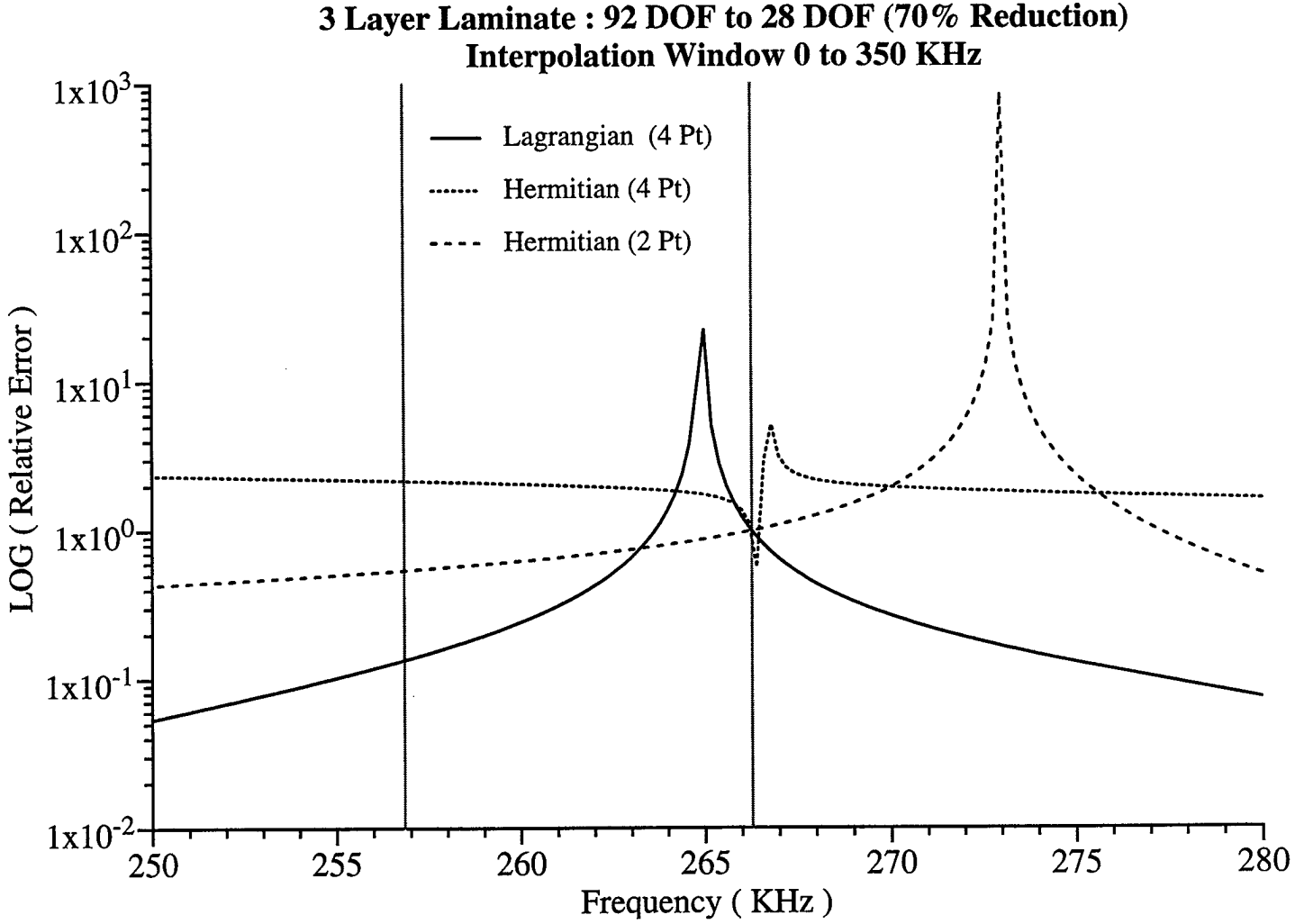


Figure 8.1.6 – Relative Error comparing different interpolation functions for the interpolation window range of 0 to 350 KHz and a frequency sweep of 250 to 280 KHz for a frequency step size of 150 Hz.

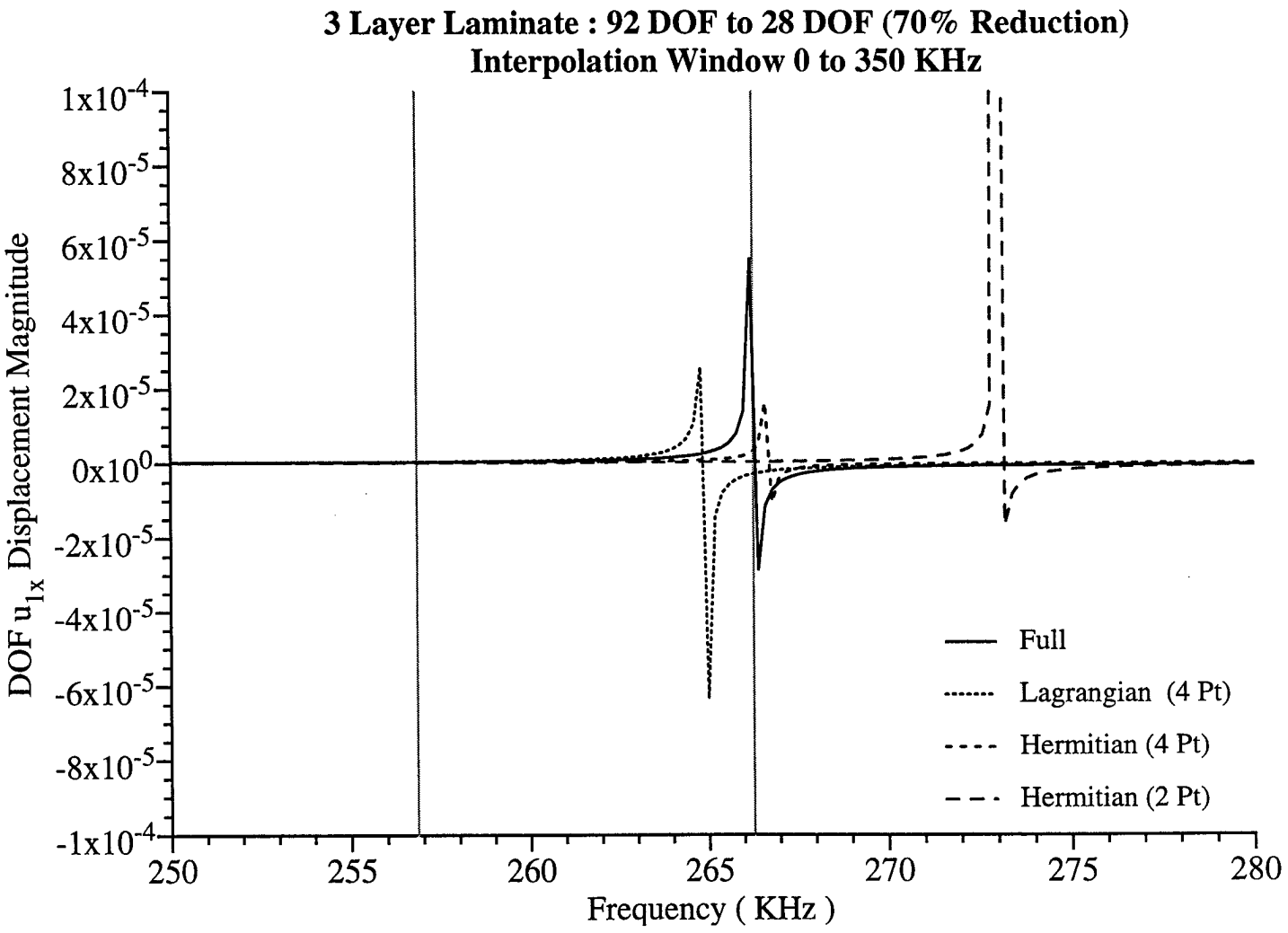


Figure 8.1.7 – The "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing interpolation functions for the window of 0 to 350 KHz and a frequency sweep of 250 to 280 KHz at a step size of 150 Hz.

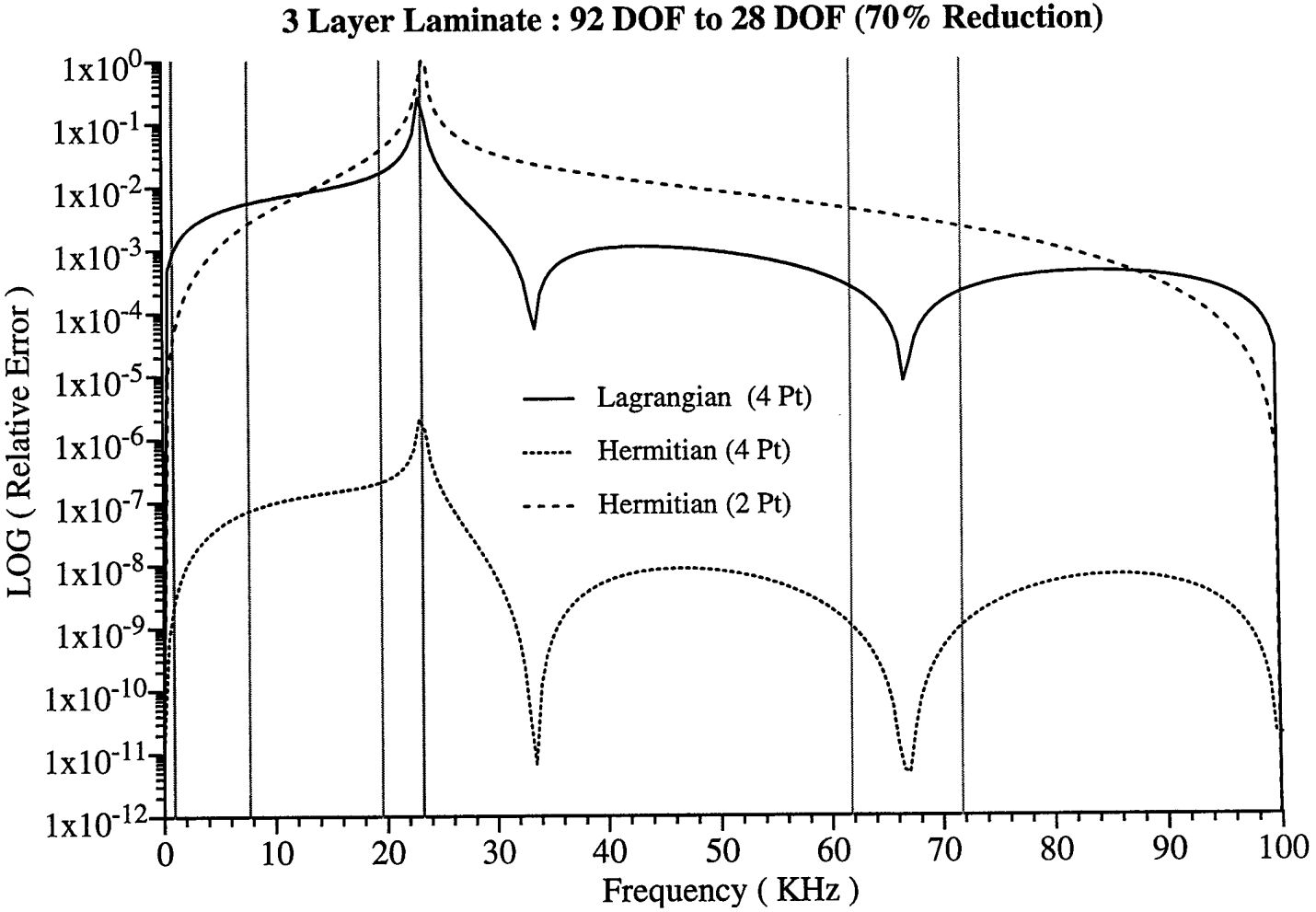


Figure 8.1.8 – Relative Error comparing interpolation functions for the frequency range of 0 to 100 KHz and a frequency step size of 500 Hz.

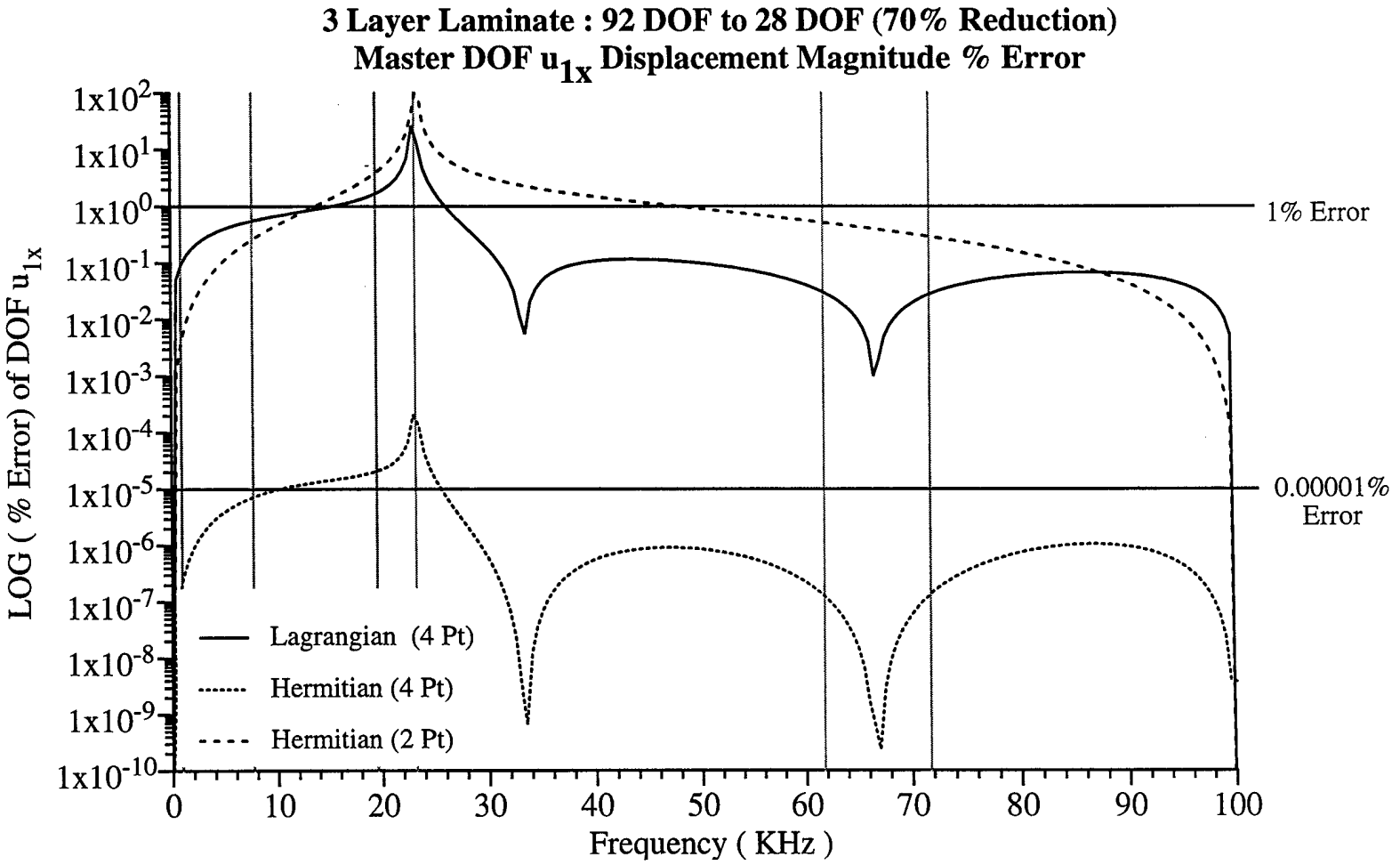


Figure 8.1.9 – % Error for the "master" DOF $|u|_{1x}$ comparing interpolation functions for a frequency range of 0 to 100 KHz at a frequency step size of 500 Hz.

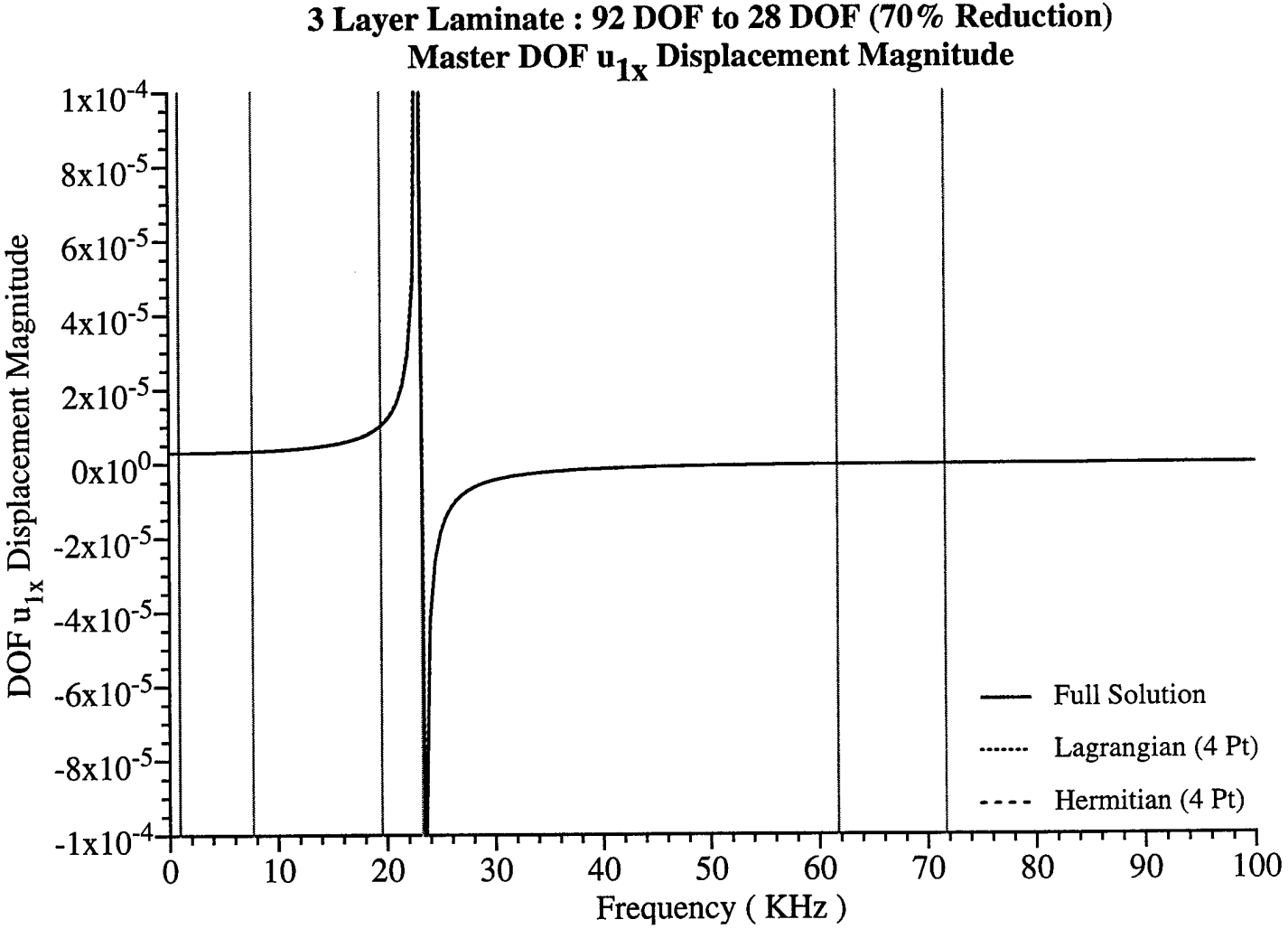


Figure 8.1.10 – The "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing the 4pt Lagrangian and the 4pt Hermitian interpolation functions to the full DOF solution for the frequency range of 0 to 100 KHz.

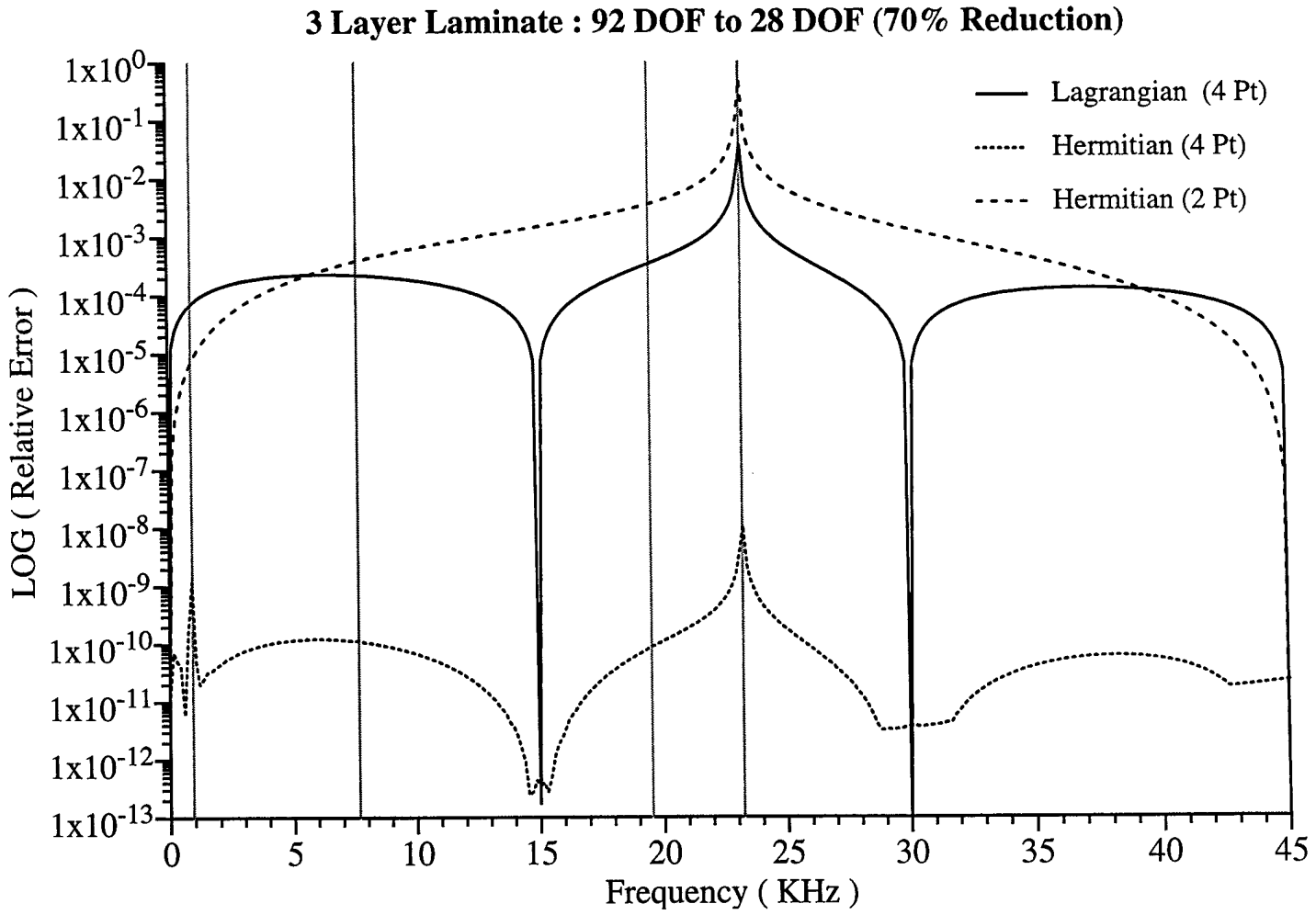


Figure 8.1.11 – Relative Error comparing interpolation functions for the frequency range of 0 to 45 KHz and a frequency step size of 150 Hz. NOTE: 70% DOF Reduction.

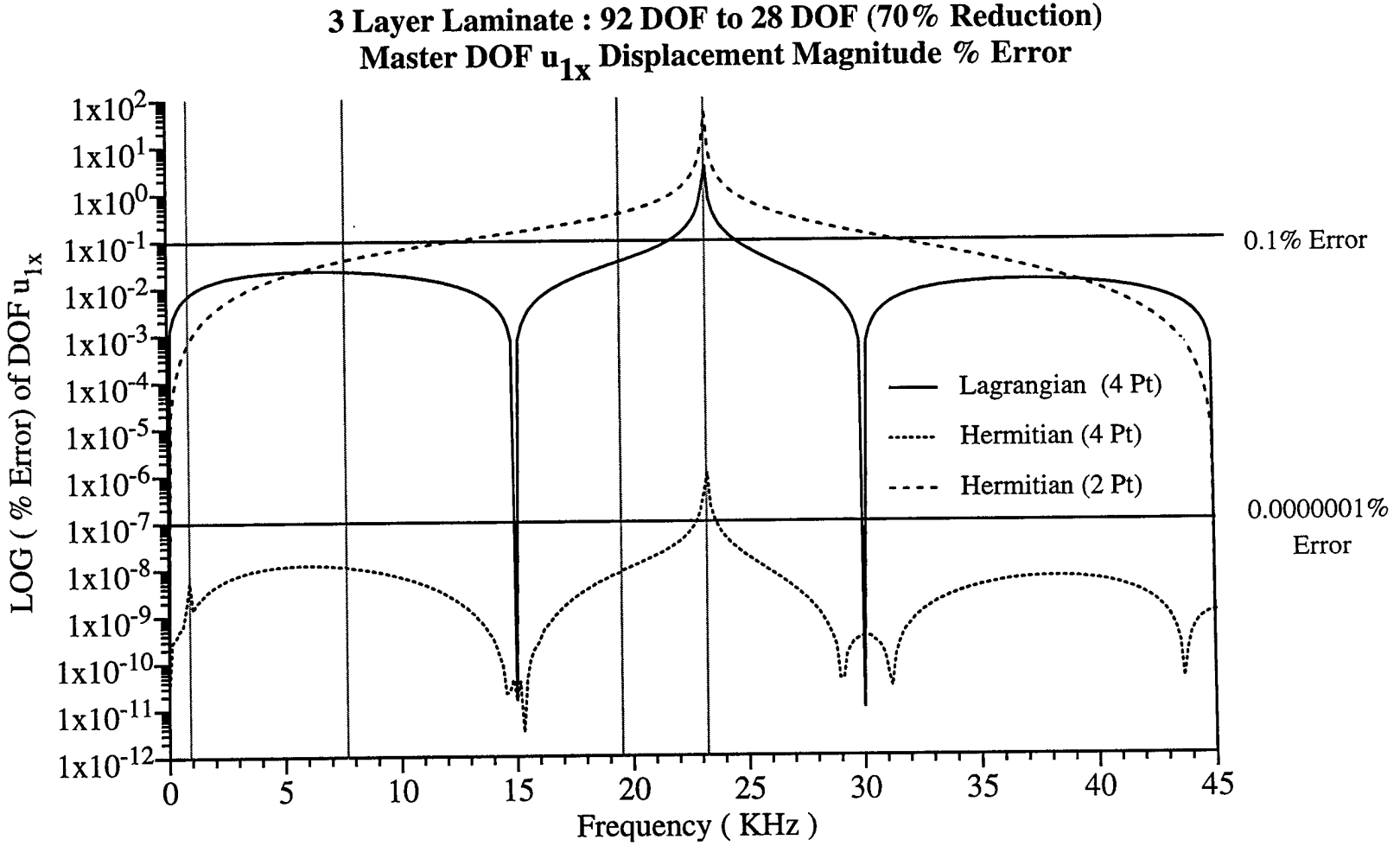


Figure 8.1.12 – % Error for the "master" DOF $|u|_{1x}$ comparing interpolation functions for a frequency range of 0 to 45 KHz at a frequency step size of 150 Hz. NOTE: 70% DOF Reduction.

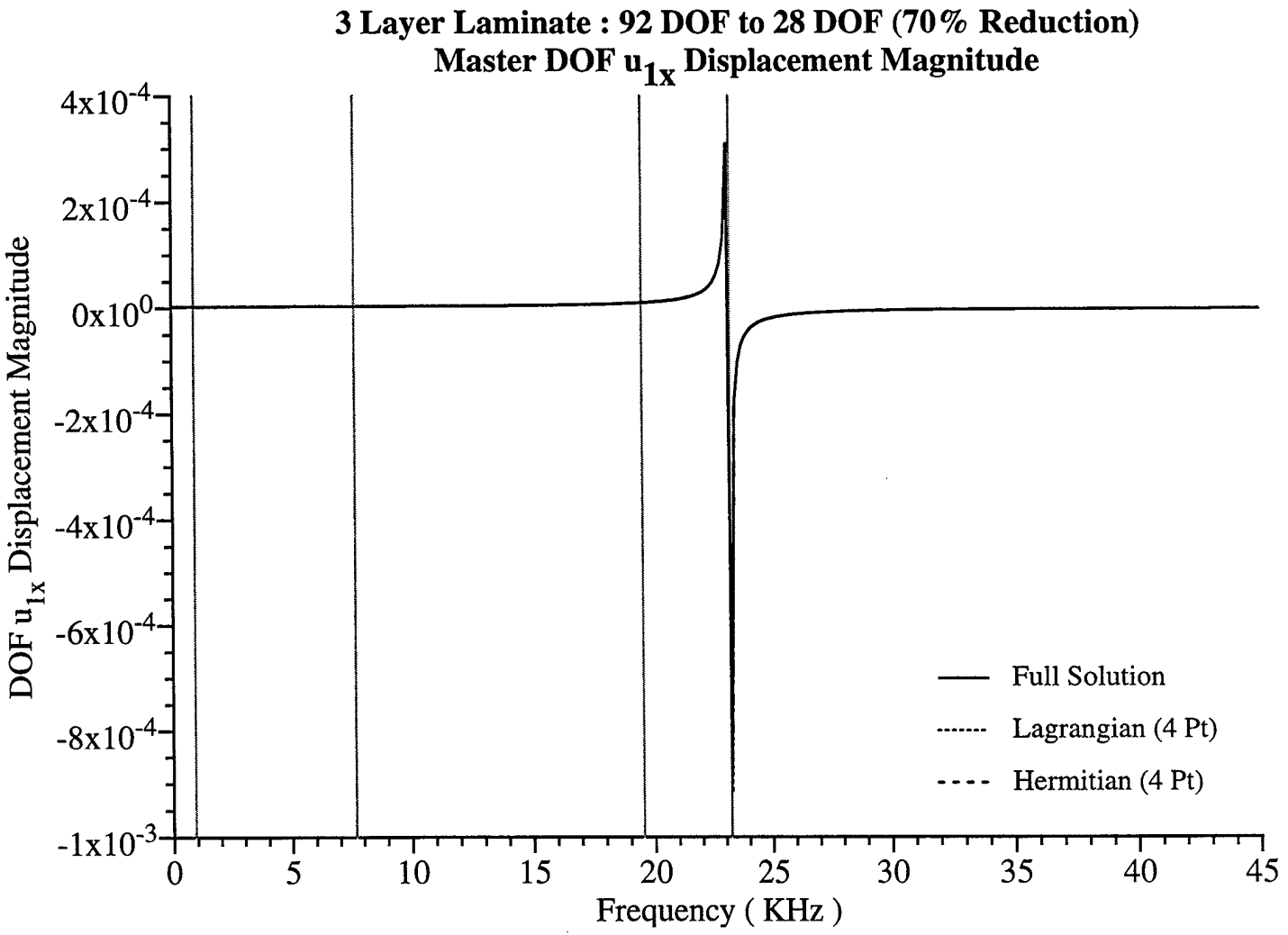


Figure 8.1.13 – The "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing two interpolation functions to the full DOF solution for a frequency range of 0 to 45 KHz. NOTE: 70% DOF Reduction.

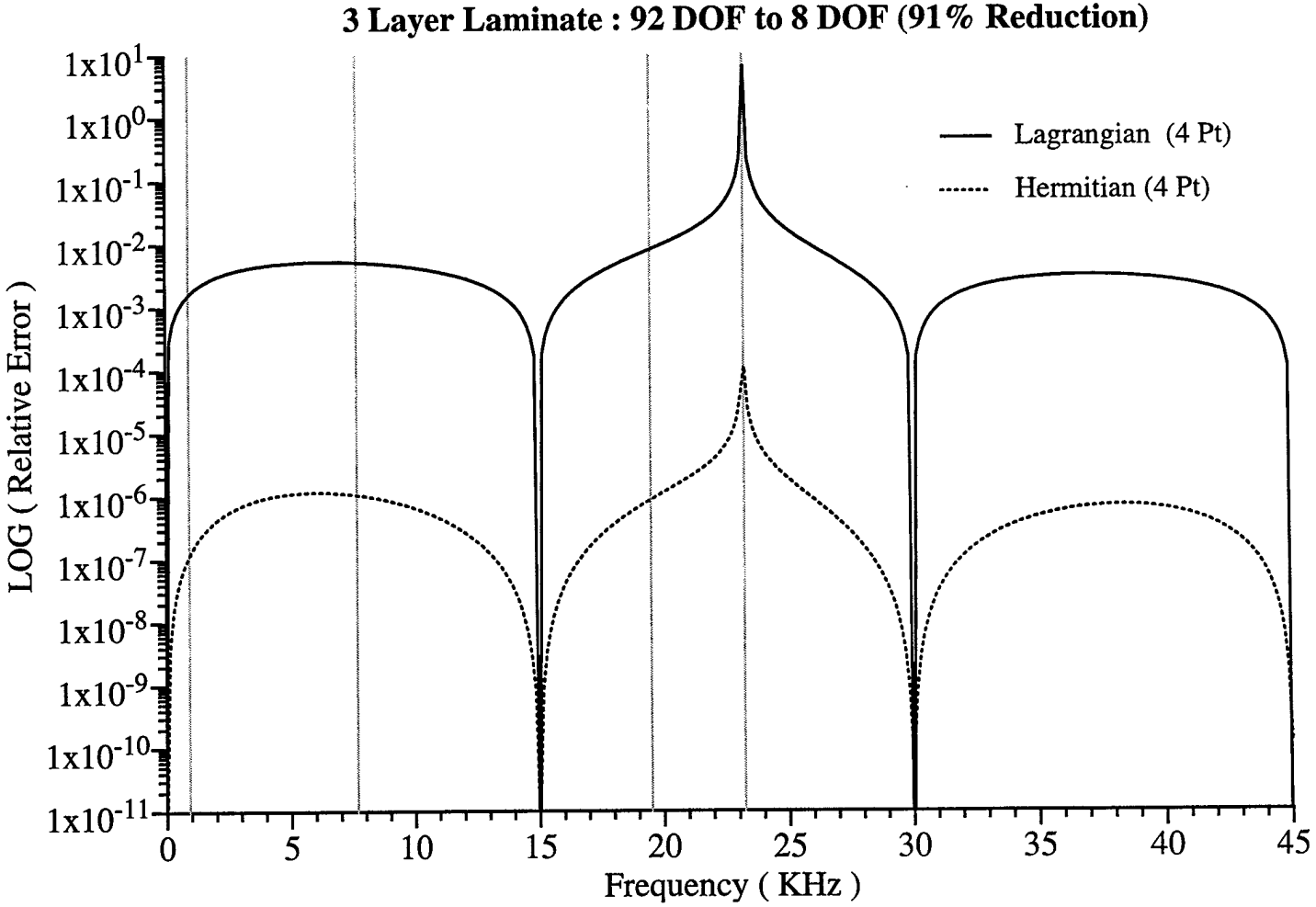


Figure 8.1.14 – Relative Error comparing the 4pt Lagrangian and the 4pt Hermitian interpolation methods for the frequency range of 0 to 45 KHz and a frequency step size of 150 Hz. NOTE: 91% DOF Reduction.

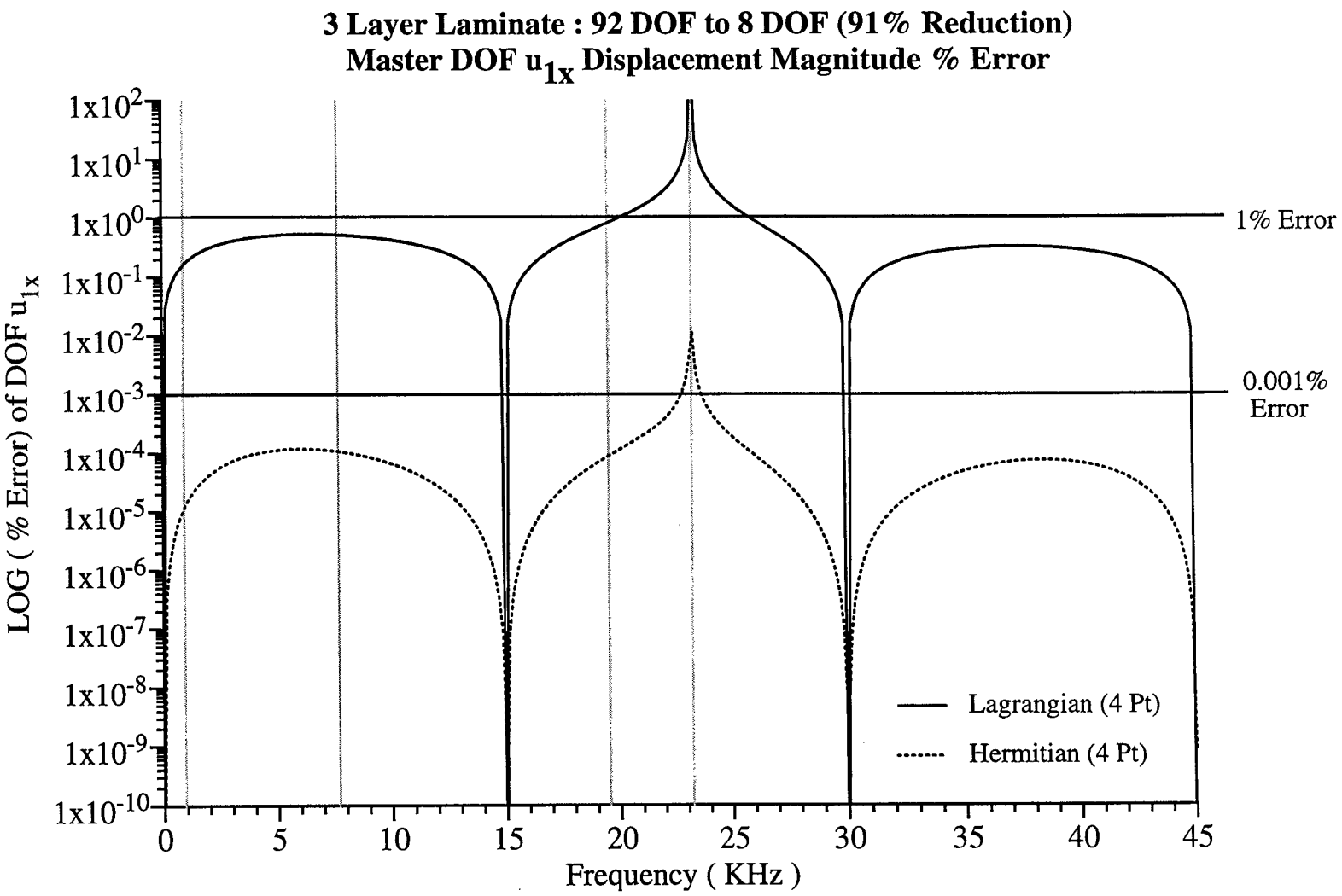


Figure 8.1.15 – % Error for the "master" DOF $|u|_{1x}$ comparing the interpolation methods for the frequency range of 0 to 45 KHz and a frequency step size of 150 Hz. NOTE: 91% DOF Reduction.

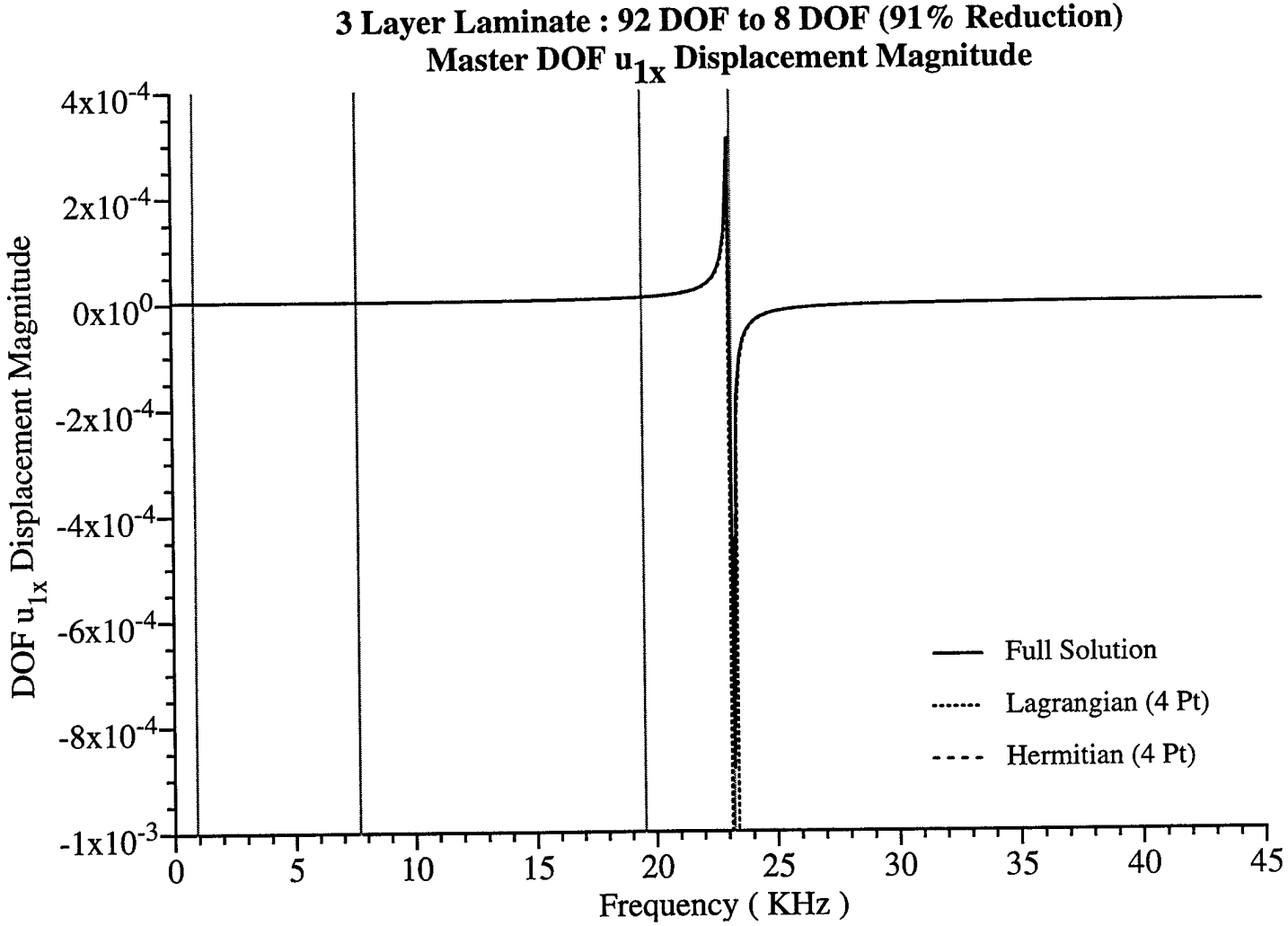


Figure 8.1.16 – The "master" DOF $|u|_{1x}$ comparing the interpolation methods for the frequency range of 0 to 45 KHz and a frequency step size of 150 Hz. NOTE: 91% DOF Reduction.

8.2 3-Layer Steel-Polystyrene 2D Model

A two-dimensional (2D) model based on the above 1D model of the 3-layer laminate was considered next. The FEM model was formed from 88 ABAQUS™ 2D plain stress elements consisting of 115 node points representing 230 DOFs (2 degrees of freedom per node point) Figure 8.2.1. Thirty-five "master" nodes (70 DOFs) were chosen, as above, and are shown in the figure as "•" symbols. This represents a 70% reduction in the degrees of freedom, i.e. 230 DOFs for the full problem to 70 DOFs.

Initially an interpolation frequency window of 0 to 25 KHz was examined, however it was found that this was too large of a frequency range for this model. A smaller interpolation window of 0 to 10 KHz was examined comparing the different interpolation functions. A frequency sweep step size of 50 Hz was chosen, and the relative error is shown in Figure 8.2.2. While the 4^{pt} Lagrangian and the 2^{pt} Hermitian interpolation functions show relatively large errors, the 4^{pt} Hermitian interpolation function has very low error. The comparison of the % error of the magnitude of the displacement for the "master" DOF u_{1x} is shown in Figure 8.2.3. The 4^{pt} Lagrangian and the 2^{pt} Hermitian interpolation methods show very large errors (approximately 10%) compared to the full DOF solution in Figure 8.2.4. Where as the 4^{pt} Hermitian interpolation shows a surprisingly low error of approximately 0.01%. All the interpolation methods seem to capture the eigenvalues that are found within the frequency window, Table 8.2.1.

Table 8.2.1 - Calculated Constrained Eigenvalues

Frequency KHz
0.2652
2.0918
2.1092
4.6078
11.6655
15.2256
16.4790
18.9294
22.0093
23.1288

From both Figures 8.2.2 and 8.2.3, it can be seen that errors are higher in the low frequency range and decrease across the frequency window. This fact arises from the presence of an eigenvalue at the very low end of the window. The presence of this eigenvalue has an effect on the smoothness of the interpolated function in the low end of the frequency window. This shows that the interpolated function is sensitive to the eigenvalues of the system particularly if they are at or near the end interpolation (inversion) points. Several options are available to improve the over all accuracy; shifting of the frequency interpolation window away from the end point eigenvalues, the reduction of the frequency window range or variable placement the interior interpolation (inversion) points.

It is possible within the FWR codes to vary the placement of the interior interpolation (inversion) points for both the Lagrangian and Hermitian interpolation methods. The above frequency window was examined by placing the interior points at 1.5 KHz and 7.5 KHz (Figures 8.2.5 to 8.2.7) and at 1.5 KHz and 4 KHz (Figures 8.2.8 to 8.2.10). While the movement of the interior points does not generally reduce the *overall* error, it does however tend to smooth the error more uniformly across the entire window. This can most readily be seen in the % error of the magnitude of the displacement for the "master" DOF u_{1x} in Figures 8.2.6 and 8.2.9 as compared to Figure 8.2.3 (interior points at 3.3 KHz and 6.7 KHz). It can be seen in all three figures, that the % error seems to be apparently the same at 1% and 0.001% for the 4^{pt} Lagrangian and for the 4^{pt} Hermitian interpolation functions respectively. However it is clear that the variable placement of the interior point does effect the error in different smaller local frequency regions within the window. There is apparently no significant effect on the actual solution and its accuracy, i.e. magnitude of the displacement for a particular node point.

A further reduced frequency window range, 0 to 5 KHz, was examined for a frequency sweep step size of 25 Hz. This reduced window both improves the over all accuracy as well as reduces the interpolation error associated with the eigenvalue at the lower frequency interpolation point, Figure 8.2.11. The slightly higher error at the low frequencies is still present but is clearly reduced. The % error of the magnitude of the displacement of DOF u_{1x} shown in Figure 8.2.12 also has reduced error for both the 4^{pt} Lagrangian and the 2^{pt} Hermitian interpolation functions to approximately 1% and 4^{pt} Hermitian interpolation function has a much lower error at

0.00001%. As seen in Figure 8.2.13, the magnitude of the displacement of DOF u_{1x} now compares extremely well to the full DOF solution for both the 4^{pt} Lagrangian and the 4^{pt} Hermitian interpolation functions.

3-Layer Steel-Polystyrene 2D Figures

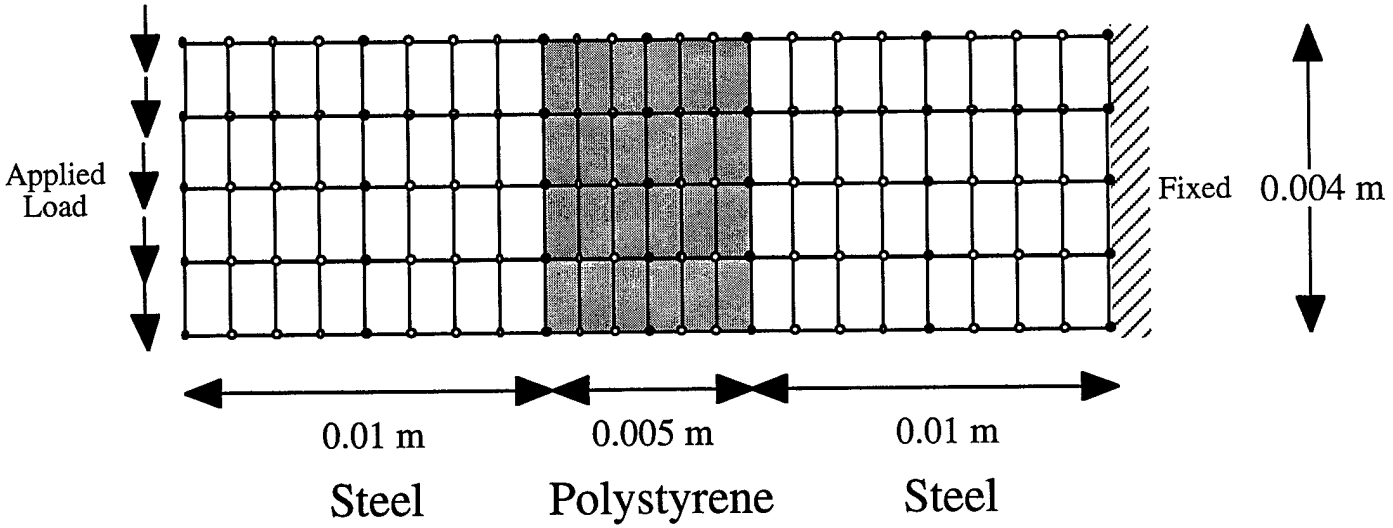


Figure 8.2.1 – 3-Layer Steel-Polystyrene Laminate 2D Model

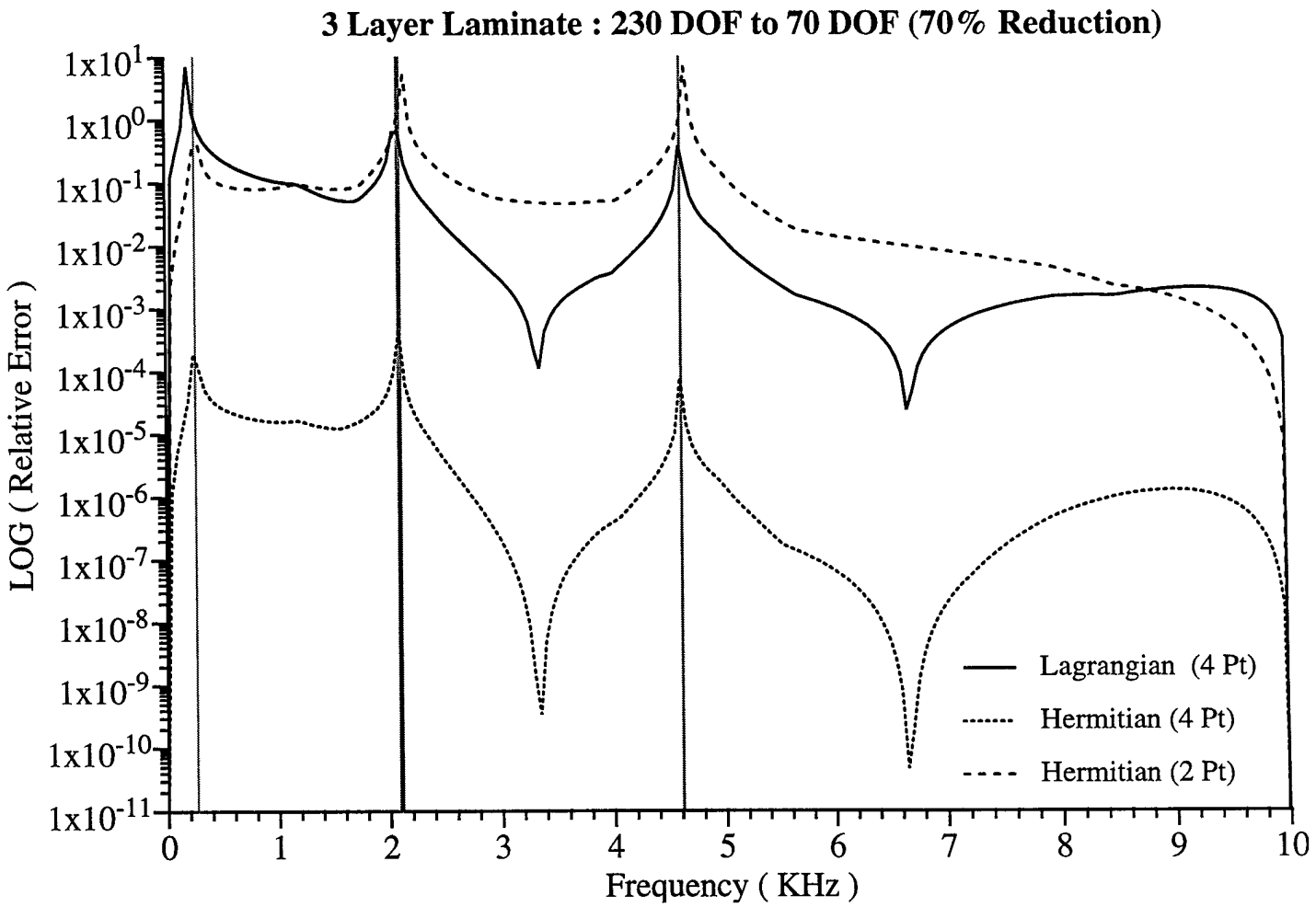


Figure 8.2.2 – Relative Error comparing different interpolation functions for the frequency range of 0 to 10 KHz and a frequency step size of 50 Hz.

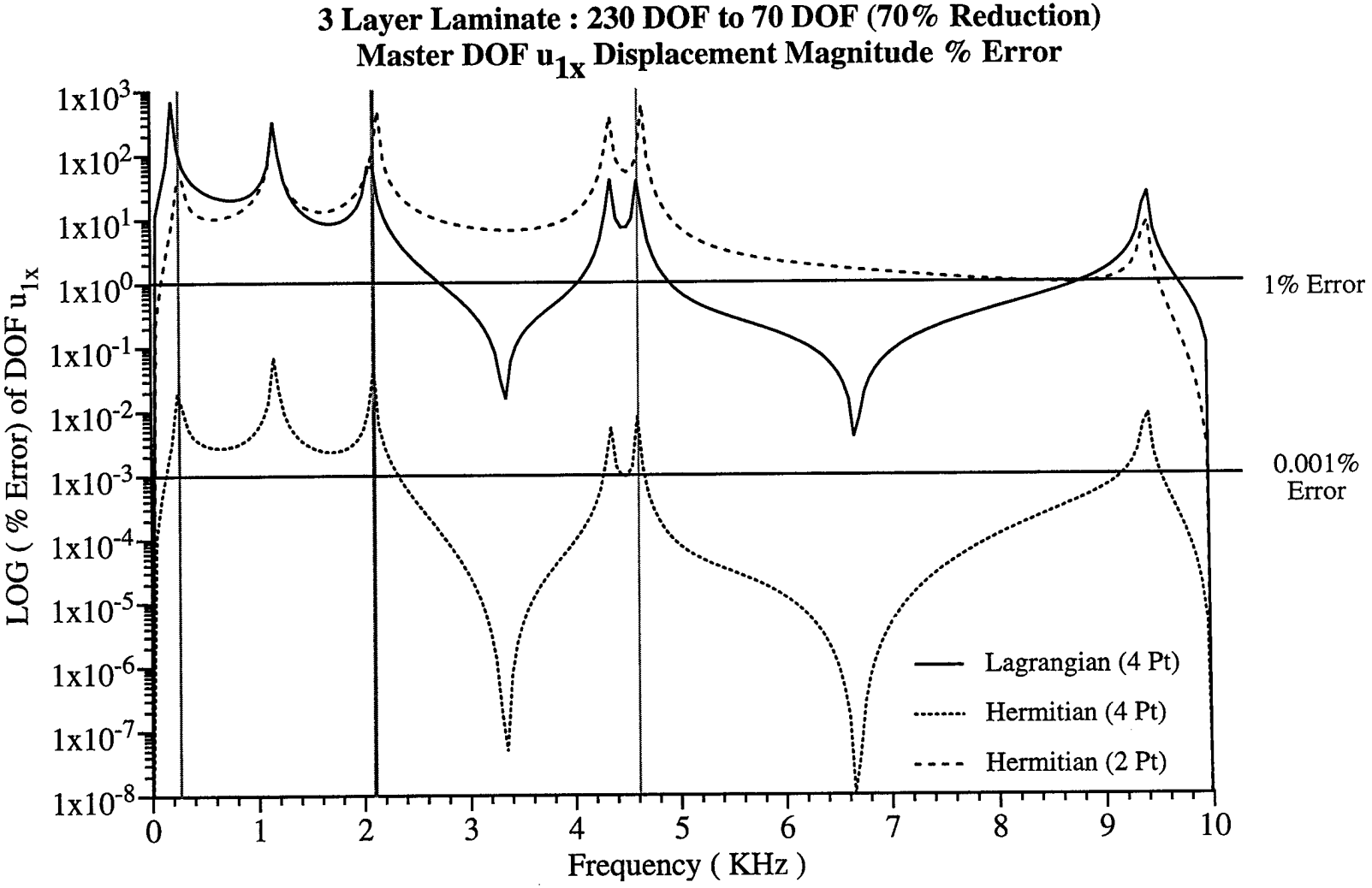


Figure 8.2.3 – % Error for the "master" DOF $|u|_{1x}$-comparing interpolations for the frequency range of 0 to 10 KHz and a step size of 50 Hz. NOTE: Interior interpolation (inversion) points are at 3.3 KHz and 6.7 KHz.

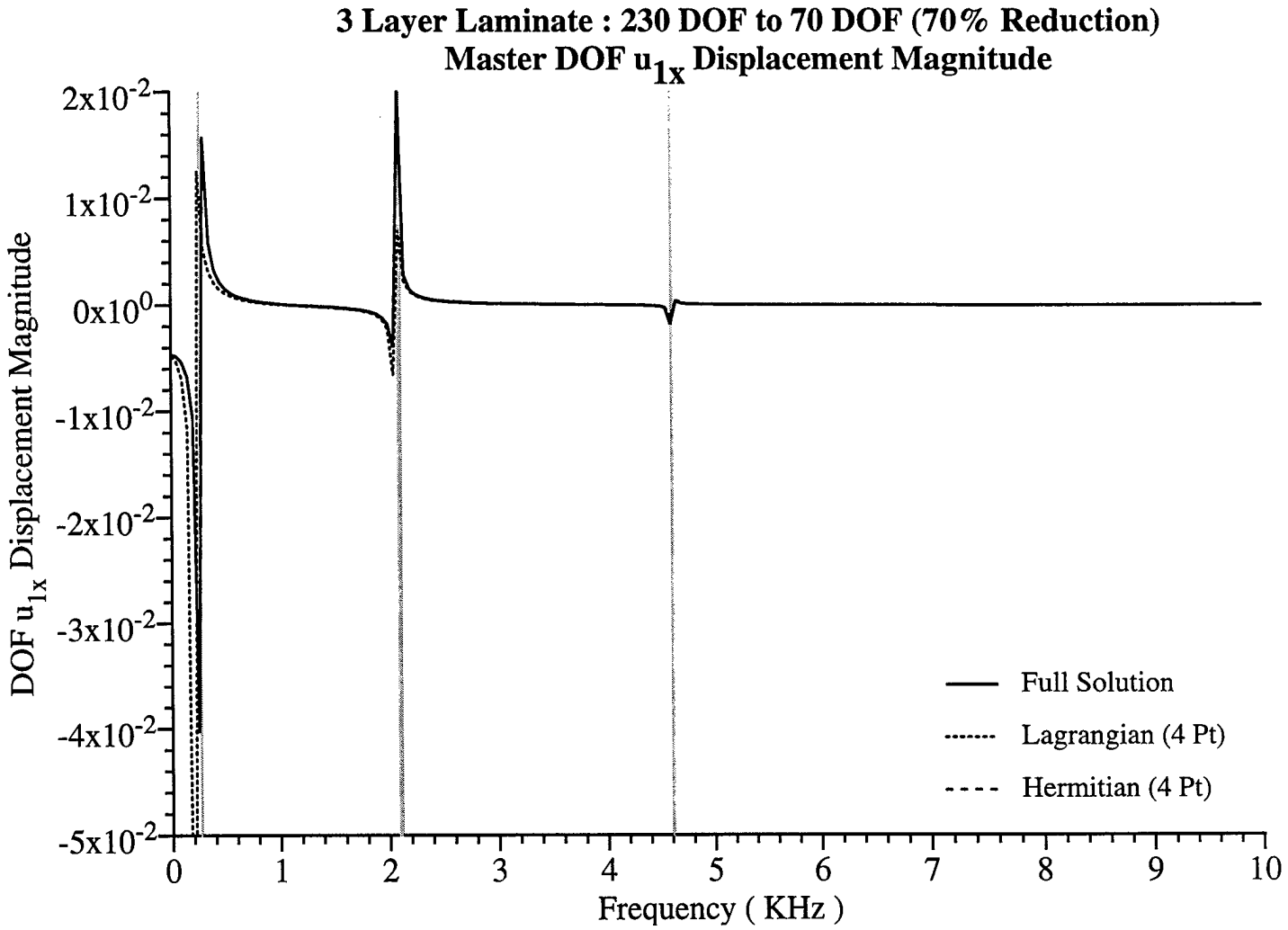


Figure 8.2.4 – The "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing two interpolation functions to the full DOF solution for the frequency range of 0 to 10 KHz.

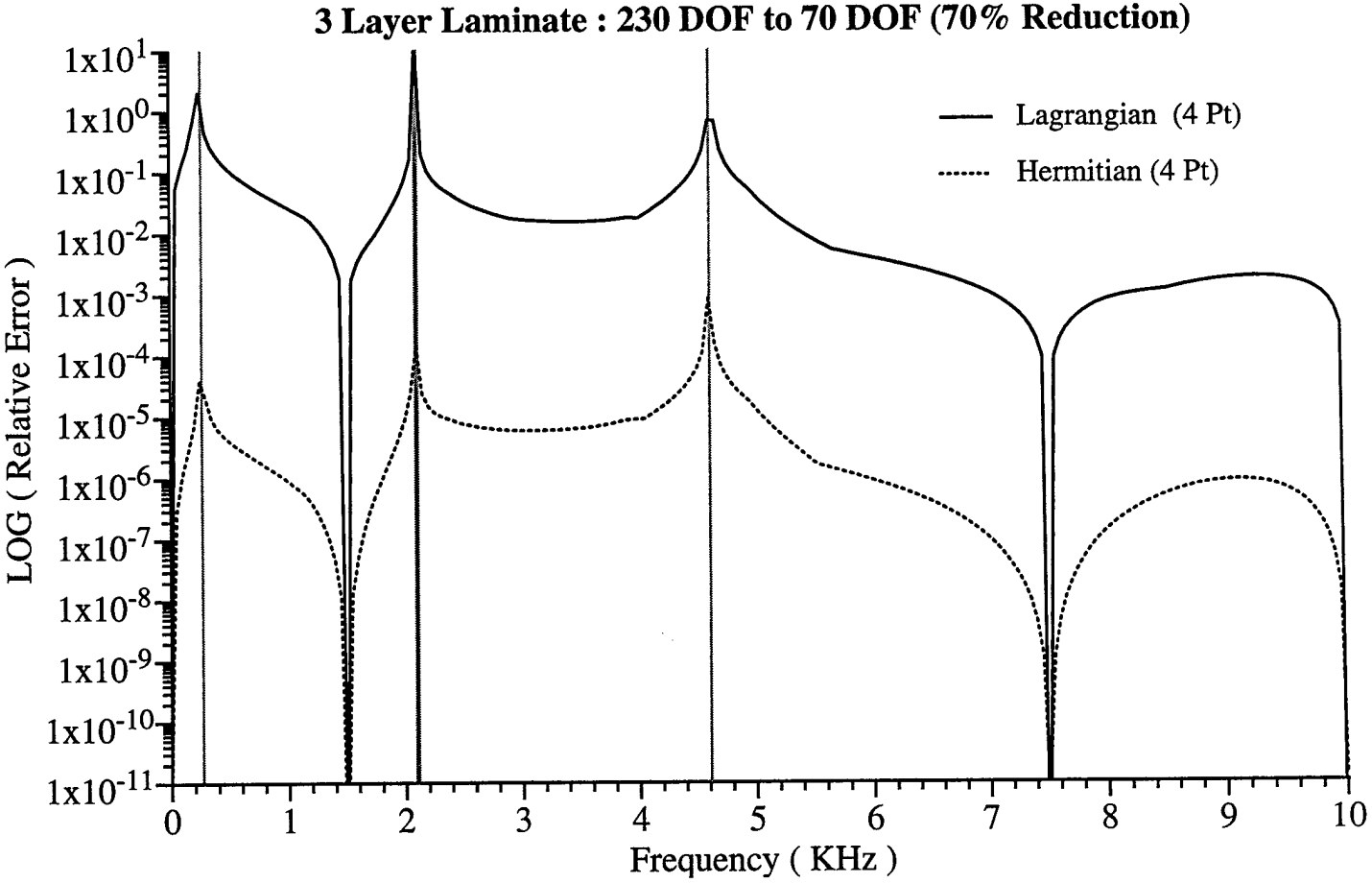


Figure 8.2.5 – Relative Error comparing interpolation functions for the range of 0 to 10 KHz and a frequency step size of 50 Hz. NOTE: Interior interpolation (inversion) points are placed at 1.5 KHz and 7.5 KHz.

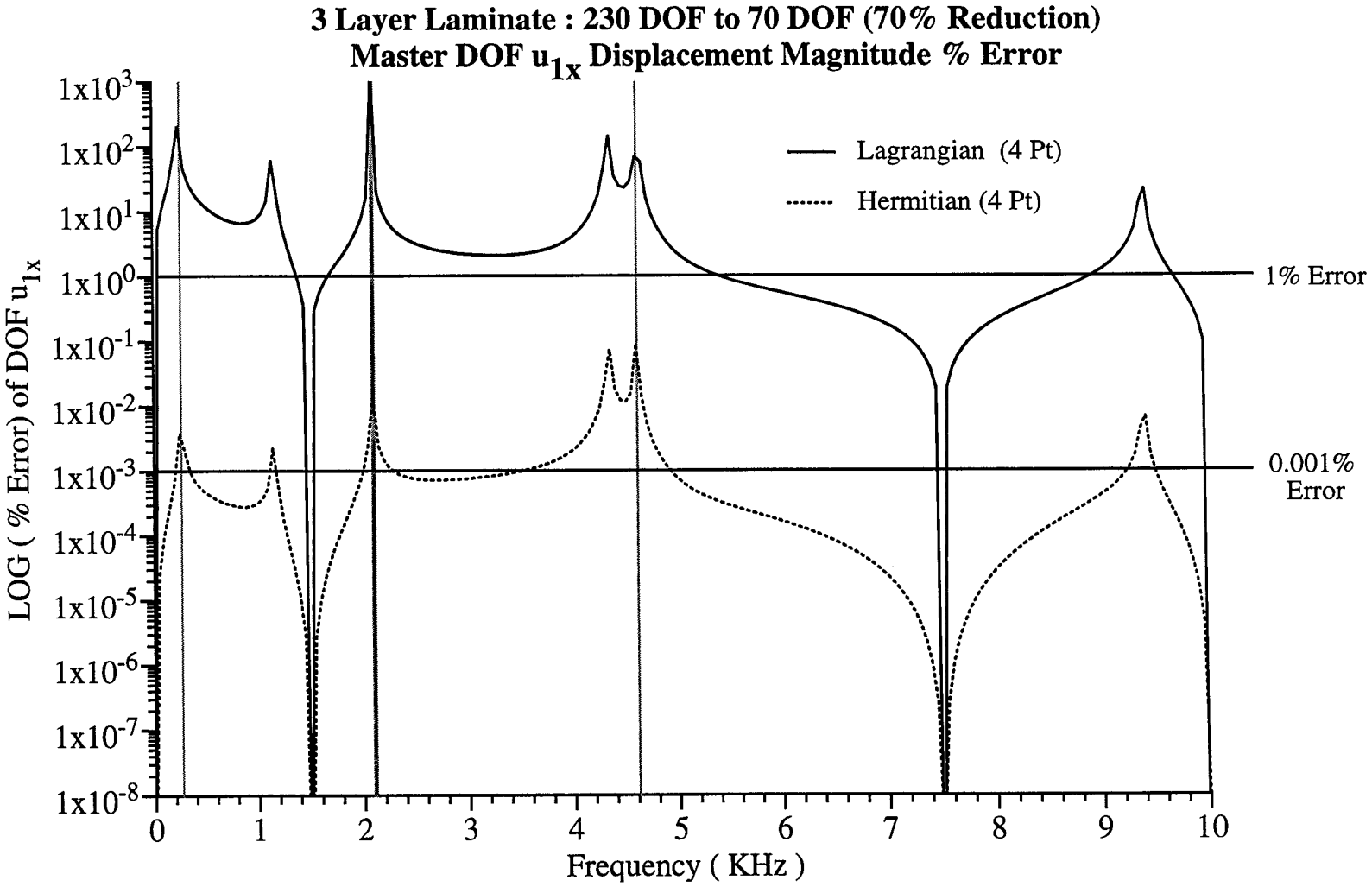


Figure 8.2.6 – % Error for the "master" DOF $|u|_{1x}$-comparing interpolations for the range of 0 to 10 KHz and a step size of 50 Hz. NOTE: Interior interpolation (inversion) points are placed at 1.5 KHz and 7.5 KHz.

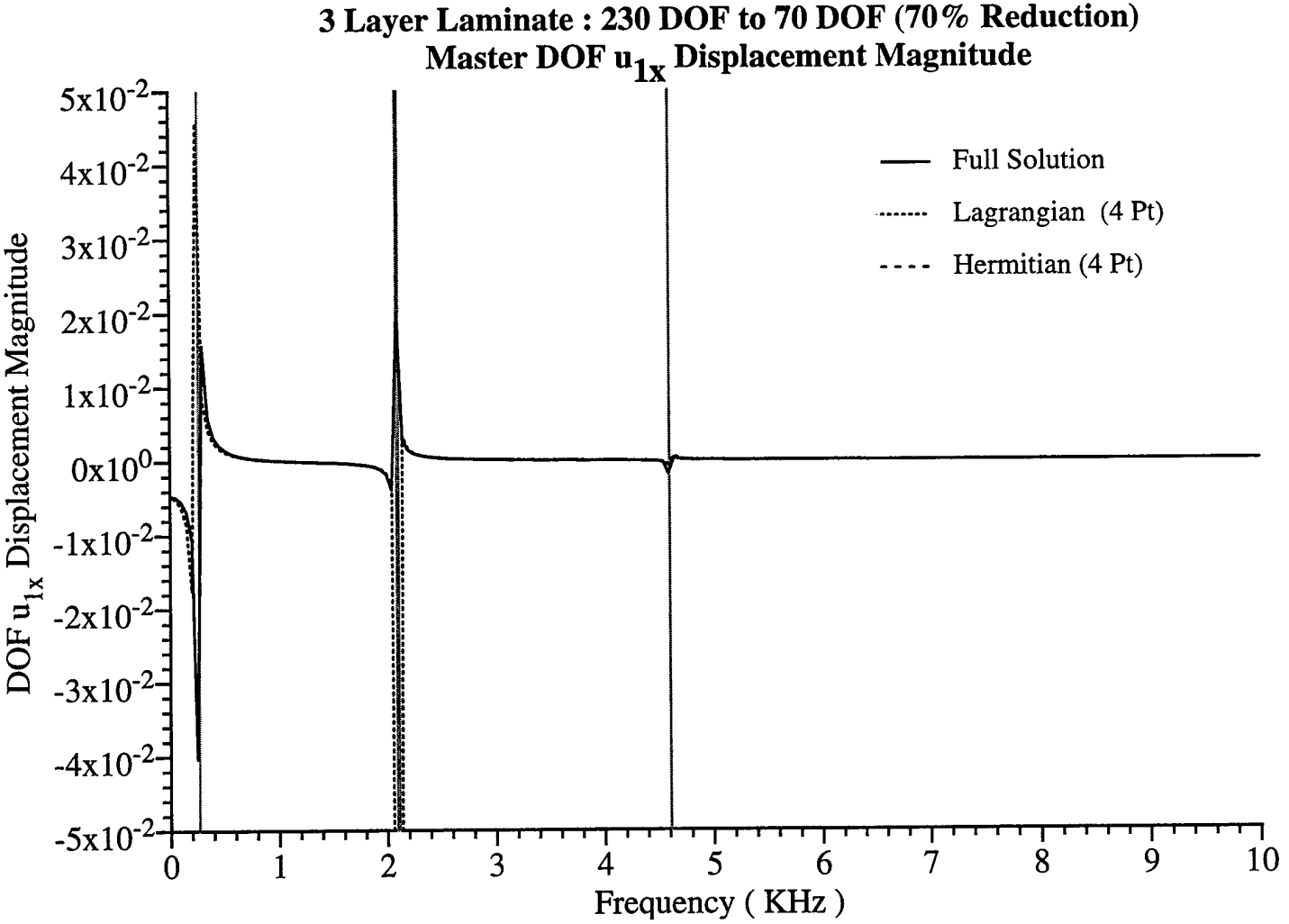


Figure 8.2.7 – The "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing the two interpolations to the full DOF solution. NOTE: Interior interpolation (inversion) points are placed at 1.5 KHz and 7.5 KHz.

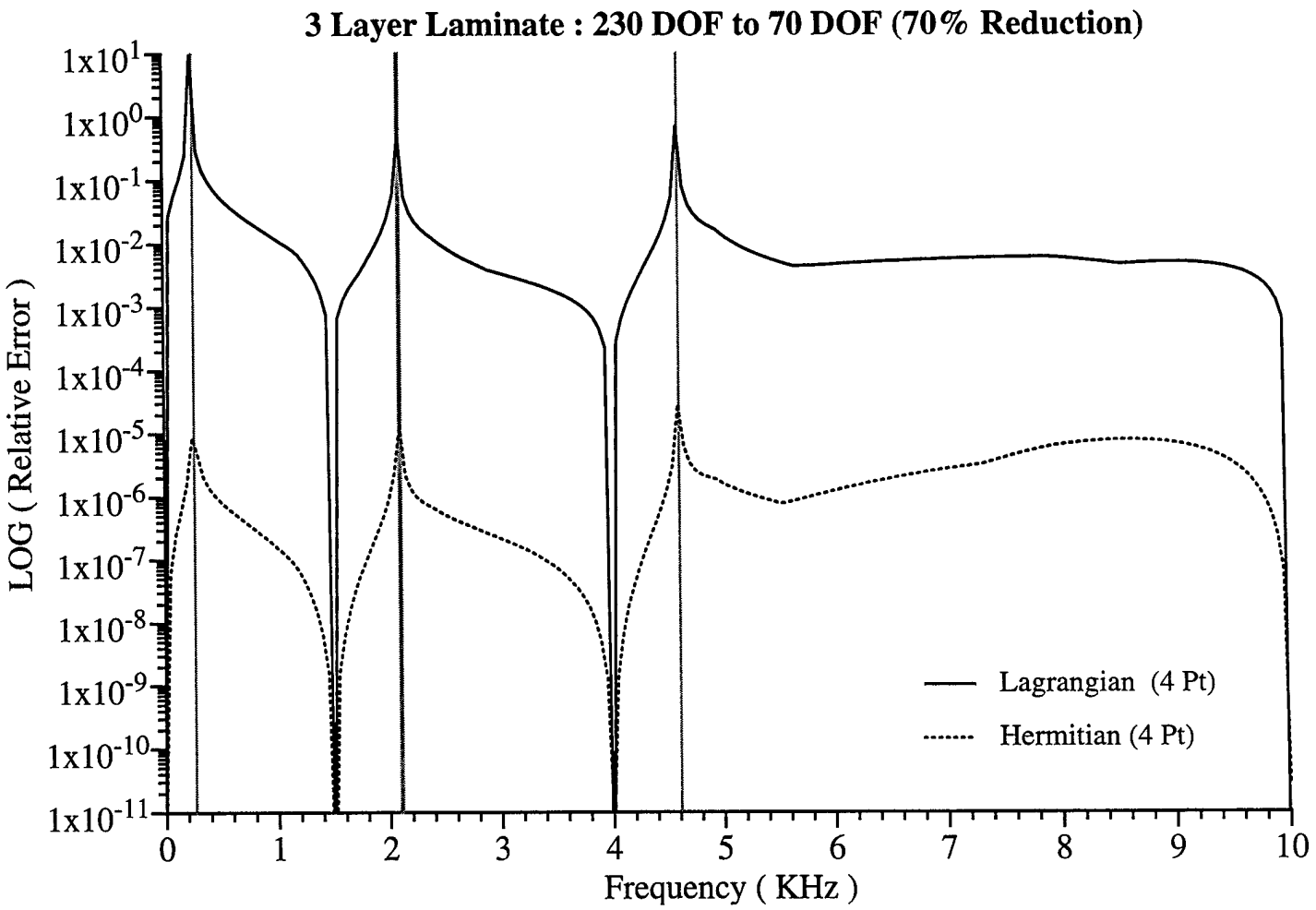


Figure 8.2.8 – Relative Error comparing interpolation functions for the range of 0 to 10 KHz and a frequency step size of 50 Hz. NOTE: Interior interpolation (inversion) points are placed at 1.5 KHz and 4 KHz.

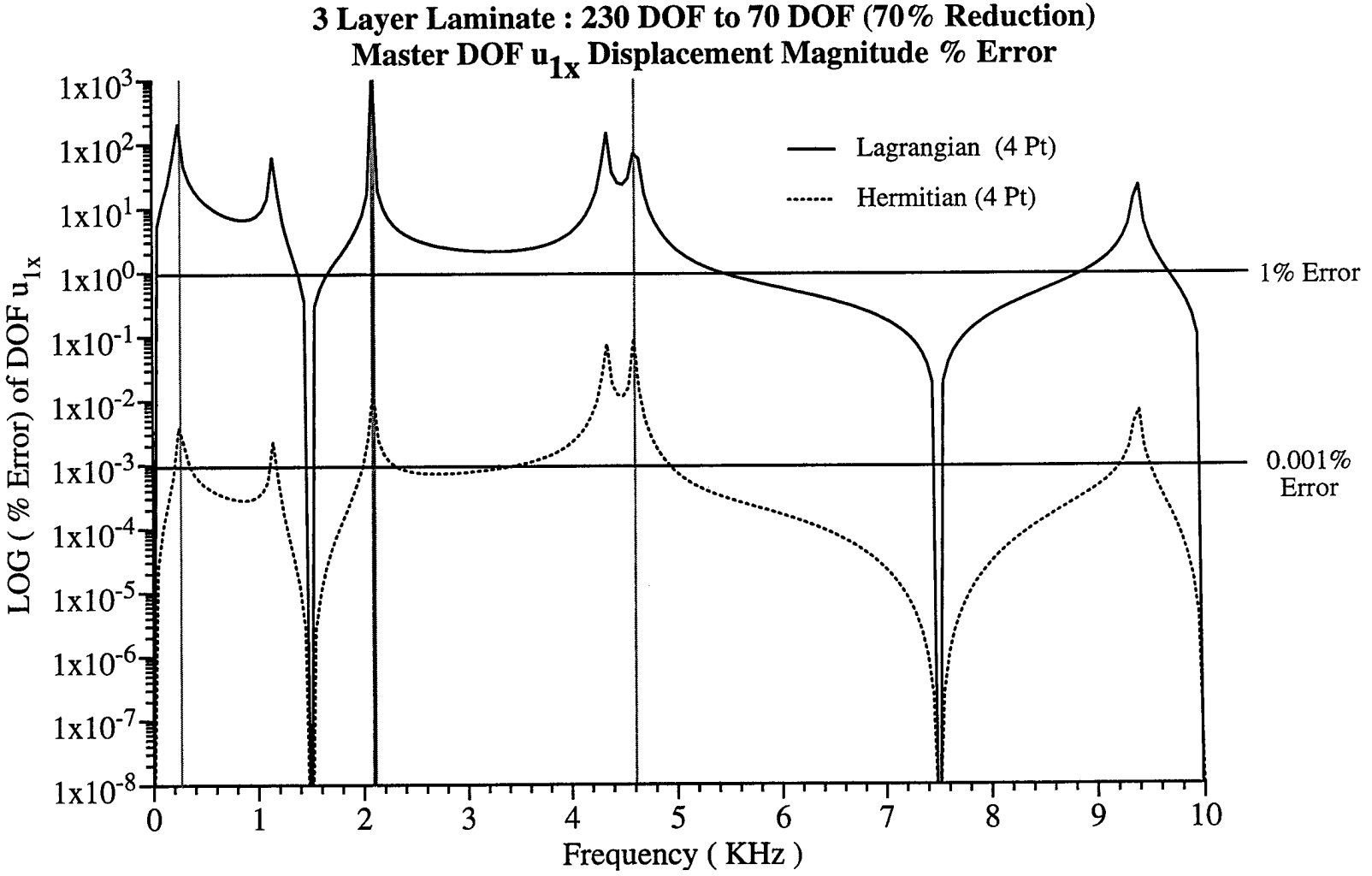


Figure 8.2.9 – % Error for the "master" DOF $|u|_{1x}$-comparing interpolation functions for the range of 0 to 10 KHz and a step size of 50 Hz. NOTE: Interior interpolation (inversion) points are at 1.5 KHz and 4 KHz.

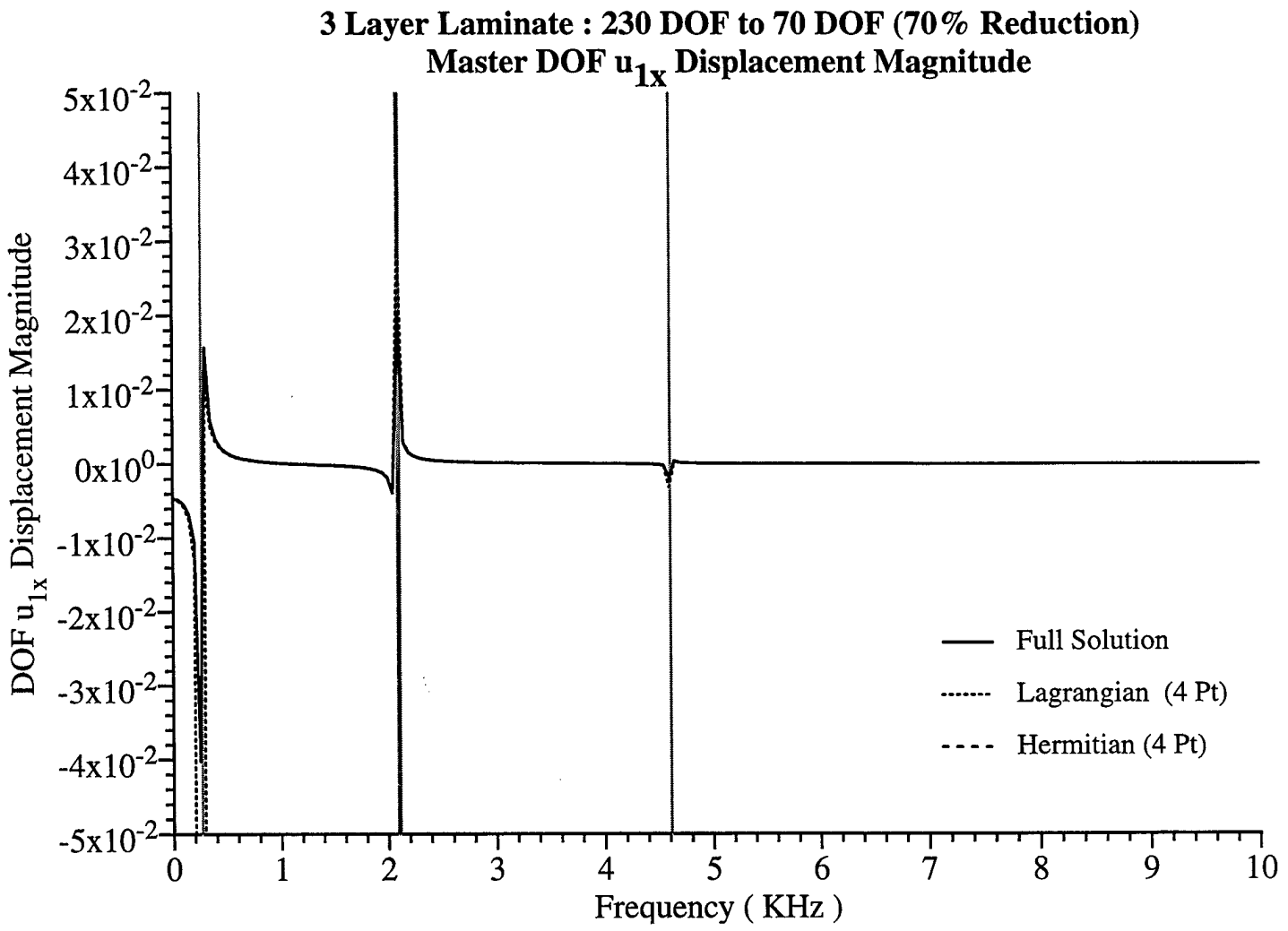


Figure 8.2.10 – The "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing the two interpolations to the full DOF solution. NOTE: Interior interpolation (inversion) points are placed at 1.5 KHz and 4 KHz.

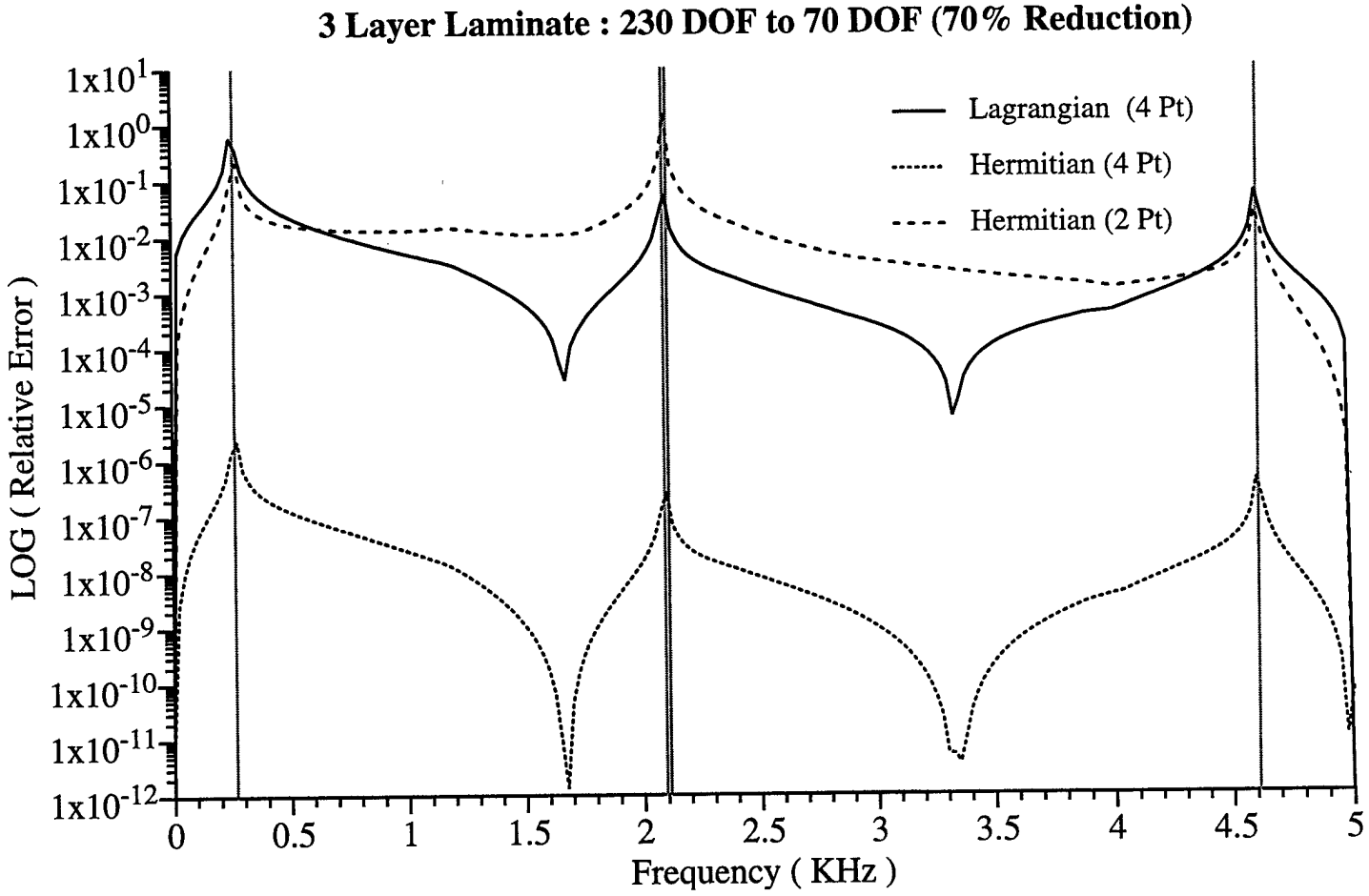


Figure 8.2.11 – Relative Error comparing interpolations for the range of 0 to 5 KHz and a step size of 25 Hz.

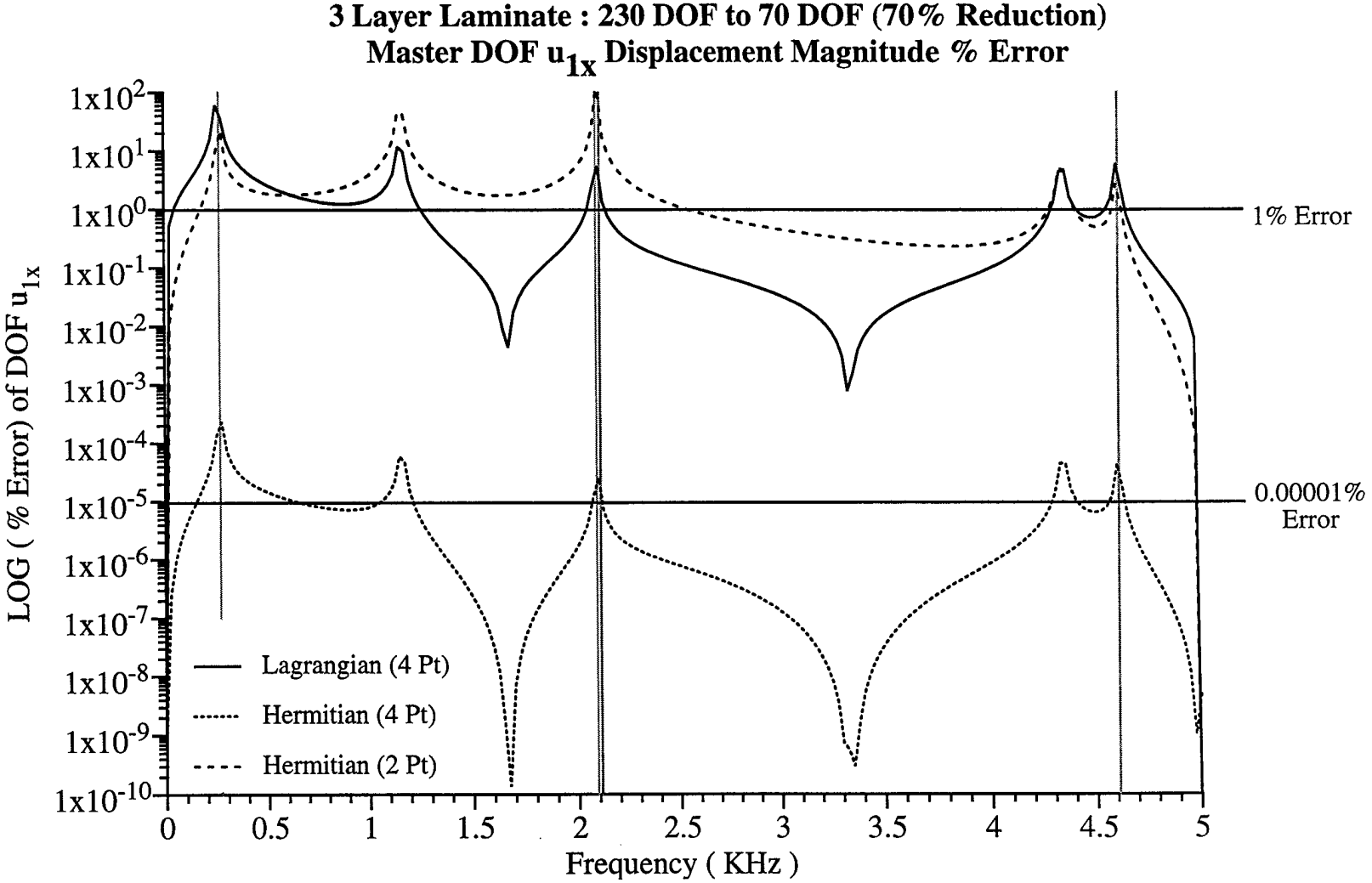


Figure 8.2.12 – % Error for the "master" DOF $|u|_{1x}$ comparing interpolations for the frequency range of 0 to 5 KHz and a frequency step size of 25 Hz.

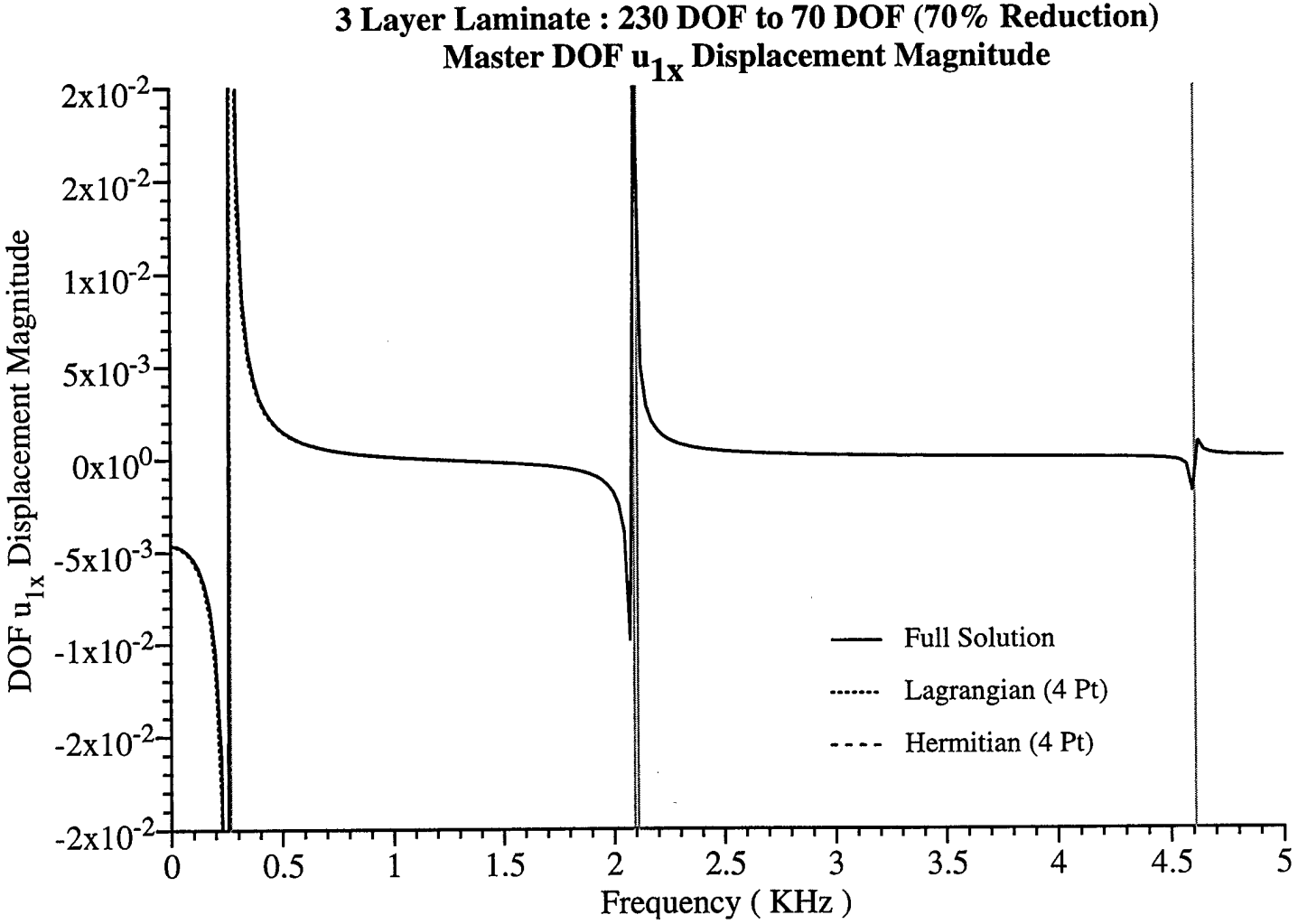


Figure 8.2.13 – The "master" DOF $|u|_{1x}$ (magnitude of the displacement) comparing the 4pt Lagrangian and the 4pt Hermitian interpolation function to the full DOF solution for the frequency range of 0 to 5 KHz.

8.3 Spherical Half-Shell Model

The next problem considered is a simple spherical shell as shown in Figure 8.3.1 with the properties listed in Table 8.3.1. A similar problem is discussed in the ABAQUS™ User's Manual [10]. The model is constructed using 50 quadratic axisymmetric SAX2 isoparametric shell elements for the undamped, free vibration problem. The total number of node points is 101 with 3 degrees of freedom per node point; hence the total DOFs for this model is 303. The number of "master" node points was chosen as 29, so the degree of reduction of the model is 71%. The selected "master" node points are shown in the Figure 8.3.1 as "•" symbols (Node #s 1, 51 and 101) which represent the loaded and constrained nodes. The remaining "master" nodes were arbitrarily chosen from the model (Node #s 5, 9, 13, 17, 21, 25, 27, 29, 33 and 37).

Table 8.3.1 – Steel Half-Spherical Shell Properties

Young's Modulus	180.0 GPa
Poisson Ratio	0.333
Density	7670 Kg/m^3

Initially a large frequency interpolation window of 0 to 500 Hz was examined with a fairly coarse frequency sweep step size of 2 Hz. Only the 4^{pt} Lagrangian and the 4^{pt} Hermitian interpolation methods were compared, shown in Figure 8.3.2. As can be seen there are a large number of eigenvalues in this window. Surprisingly both interpolation functions capture the functional behavior at *all* of the eigenvalues despite the large window and coarse step size. In Figure 8.3.3 the Lagrangian interpolation function shows a larger % error of the magnitude of the displacement of the "master" node u_{51x} of approximately 10%, where as the Hermitian shows an approximate error of only 0.01%. From this figure and the actual magnitude of the displacement of the "master" node u_{51x} of Figure 8.3.4, it can clearly be seen that the functional behavior at every eigenvalue is represented. The Figures 8.3.5 and 8.3.6 show the % error of the magnitude of the displacement of another "master" node u_{101y} and its magnitude. Identical errors as well as the direct parallel response at the eigenvalues are seen as above. Figures 8.3.7 to 8.3.12 show the % error of the magnitude of the displacement and the magnitude of the displacements for a "reconstructed" "slave" node u_{75} for each of its three degrees of freedom. All of these plots demon-

strate the consistency of the error and the accuracy of the response of the reconstructed solution as compared to the full solution for this model.

The interpolation frequency window size was reduced to 0 to 200 Hz with the sweep step size to 1 Hz. Now within this particular window only two eigenvalues are present near the end points of the window. The relative error, Figure 8.3.13 and the % error of the "master" node u_{51x}, Figure 8.3.14, show a dramatic decrease and very accurate behavior at the eigenvalues. The 4^{pt} Hermitian seems to have extreme accuracy in this frequency range. The actual magnitude of the displacements for this DOF shows no difference in the reduced interpolated solutions and the full DOF solution, Figure 8.3.15. These figures were for a level of 71% DOF reduction (303 DOFs to 87 DOFs). This same window range was then examined with a higher level of DOF reduction of 303 DOFs to 29 DOFs or an 87% DOF reduction. Surprisingly the relative errors, Figure 8.3.16 and the % errors, Figure 8.3.17, change very little, increasing by approximately one order of magnitude. Careful examination of these two figures and the magnitude of the displacement in Figure 8.3.18 shows only that the 4^{pt} Lagrangian does not match exactly the eigenvalue behavior at the low range of the frequency window. However, the levels of error even for this significant DOF reduction are quite surprising at 1% for the Lagrangian and 0.0001% for the Hermitian methods.

As is seen in Figure 8.3.2 or Figure 8.3.19, there are a large number of eigenvalues within the frequency range of 200 Hz to 300 Hz. The frequency window range of 200 to 300 Hz was examined at both the 71% DOF reduction and the 87% DOF reduction level. This window range was chosen in order to characterize the behavior of the FWR method with a large number of eigenvalues with in a small frequency window. Figure 8.3.20 shows the relative error at 71% reduction for the window range and the sweep step size of 0.5 Hz. Both interpolation methods result in very low % errors ($\approx$0.1%) and highly accurate responses as compare to the full DOF solution, Figures 8.3.21 and 8.3.22. The characterization at 87% DOF reduction is shown in Figures 8.3.23 to 8.3.25. These figures show identical trends as described above. It should be noted that as in the above cases discussed, *all* eigenvalue responses are clearly captured in both DOF reductions.

An important fact can be observed with respect to the 4[pt] Hermitian interpolation curves in these figures. This is that the end interpolation (inversion) points of the frequency window do not approach a nearly zero value, as does the Lagrangian interpolation. It is important to note that the Hermitian interpolation method includes derivative information at all of the interpolation points. As a result if an eigenvalue is near the interpolation point particularly at the frequency window end points, the function and its derivative are rapidly varying near the interpolation points in region of validity for the interpolation function. For this window range of 200 Hz to 300 Hz, it can be seen from Table 8.3.2 (or Figure 8.3.19) that there are eigenvalues that lie just outside of the interpolation window range (188.14 Hz and 308.71 Hz). Hence the influence from these eigenvalues, even though outside of the interpolation region, does in fact contribute to the smoothness of the interpolation function and, hence, to the accuracy of the response. This shows that the continuity of the interpolation function will be maintained across the interpolation window end points. The Lagrangian value should always be nearly zero at an interpolation point.

Table 8.3.2 - Calculated Eigenvalues

Unconstrained	Constrained
187.36 Hz	188.14 Hz
222.69 Hz	223.48 Hz
236.95 Hz	237.73 Hz
244.41 Hz*	245.23 Hz
249.30 Hz	250.23 Hz
253.30 Hz*	254.44 Hz
257.26 Hz	258.75 Hz
267.02 Hz	263.68 Hz
273.53 Hz	269.63 Hz
281.49 Hz	276.87 Hz
291.12 Hz	285.68 Hz
302.59 Hz	296.24 Hz
316.06 Hz	308.71 Hz

* Used for Rayleigh Damping Calculation

Another *very* important note that applies to *all* previously described examples is that the eigenvalues of the system were NOT known beforehand. Having shown how accurately the response of the reduced DOF problem can be obtained, it is clear that the eigenvalues of the full DOF system may be closely approximated from the reduced DOF model analysis. It is important to note that these eigenvalues represent the resonances of the "reduced" DOF system, which generally have been found to be not significantly different from those of the full DOF system. In

order to obtain more accurate eigenvalues of the reduced system, small frequency sweep step sizes would be necessary which could a performed by a partial frequency sweep within an interpolated window. As described above, this is where a window range is interpolated and then only a small portion of that interpolation window is swept at a very small step size. (See for example Figures 2.2.1.6 and 2.2.1.7) These features could be *extremely* useful in developing models, understanding system responses, obtaining eigenvalues and, most importantly, in guiding the analysis of large complex structural models.

One additional aspect of FWR method is that multiple frequency windows may be analyzed and these solutions can be then combined to construct a larger frequency response window. To illustrate this feature of FWR method, a frequency interpolation window of 0 to 300 Hz was analyzed with a frequency sweep step size of 1 Hz at 71% DOF reduction. These results are presented in Figures 8.3.26 to 8.3.28. The above analyses for the frequency windows of 0 to 200 Hz and 200 to 300 Hz, can be combined and compared to the full analysis for the window range of 0 to 300 Hz. The combined analyses for the window range of 0 to 200 Hz, Figures 8.3.13 to 8.3.15, with that for 200 to 300 Hz window range, Figures 8.3.20 to 8.3.22 are shown in Figures 8.3.29 to 8.3.31. It is difficult to compare the overall accuracy (error) of the combined data as compared to the full frequency range because the two analyses, which are combined, have different parameters (window and step size). It is clear that the responses of the magnitude of the displacements shown in Figure 8.3.28 (full window) and Figure 8.3.31 (combined windows) are identical. The combinations of the 0 to 200 Hz window for 87% DOF reduction (Figures 8.3.16 to 8.3.18), with that for 200 to 300 Hz window range at 71% reduction (Figures 8.3.20 to 8.3.22) are shown in Figures 8.3.32 to 8.3.34. These figures also show nearly identical response as compare to the full 0 to 300 Hz window, despite representing different degrees of reduction. Finally a comparison of all the responses of the magnitude of the displacements (Figures 8.3.28, 8.3.31 and 8.3.34) to the analysis of the original 0 to 500 Hz window, Figure 8.3.35, also demonstrates the consistency of this method. Note that the scale for Figure 8.3.35 has been redrawn.

Rayleigh damping was next introduced into the Spherical Half Shell Model. The mass and stiffness matrices were obtained from an ABAQUS™ model and reordered according to the selection of the "master" nodes. These matrices were then used to calculate the Rayleigh damp-

ing as defined as

$$\mathbf{D} \equiv \alpha\mathbf{M} + \beta\mathbf{S} \tag{8.3.1}$$

Where the terms

$$\beta = \frac{2(\omega_2\xi_2 - \omega_1\xi_1)}{\omega_2^2 - \omega_1^2} \tag{8.3.2}$$

And

$$\alpha = 2\omega_1\xi_1 - \beta\omega_1^2 = \omega_1(2\xi_1 - \beta\omega_1) \tag{8.3.3}$$

Here the terms, ξ_1 and ξ_2 are the weightings or percentages of critical damping of the select frequency modes, ω_1 and ω_2. From the calculated eigenvalues of this model listed in Table 8.3.2, the frequency modes of ω_1 = 244.41 Hz and ω_2 = 253.30 Hz were chosen as being approximately in the middle of the frequency window range of 200 Hz to 300 Hz. The calculated values, Table 8.3.3 for α and β, according to equations 8.3.2 and 8.3.3, at these selected eigenvalues are summarized for different levels of critical damping (5%, 1% and 0.5%).

Table 8.3.3 - Calculated Values for α and β at selected eigenvalues
ω_1 = 244.41 Hz **and** ω_2 = 253.30 Hz

ξ_1	ξ_2	α	β
0.05	0.05	78.1552	3.1977×10^{-5}
0.01	0.01	15.6310	6.3955×10^{-6}
0.005	0.005	7.81552	3.1977×10^{-6}

Initially 5% of critical damping was examined for this model for the frequency window range of 200 Hz to 300 Hz and a frequency sweep step of 0.25 Hz. It was found, as expected, that both real and imaginary solution responses are obtained. Plots of the relative error for the real and imaginary solutions as compared to the full solution show very small relative error, Figures 8.3.36 and 8.3.37. It is important to note that the response near the eigenvalues does not appear as in the case with no damping (Figure 8.3.20). Both the real and imaginary % error for the magnitude of the displacement of DOF u_{51x} are shown in Figures 8.3.38 and 8.3.39. These figures show an error of less than 0.01%. Comparison of this % error in the case in which damping *is* present to the case of *no* damping (Figures 8.3.20 to 8.3.22) show near the same %

error. Comparison of the real and imaginary responses of the magnitude of the displacement of the "master" DOF u_{51x} seen in Figures 8.3.40 and 8.3.41 shows that the reduced methods as indistinguishable from the full solution response. Again most notable in all of these figures is the apparent lack of influence from the eigenvalues. This observation produced some concern as it was assumed that a small fraction of critical damping (5%) should only tend to smooth and lower the peaks associated with eigenvalues.

Table 8.3.4 - Calculated Constrained Eigenvalues

No Damping	Damping (Real)
11.2585438 Hz	9.80502865 Hz
188.137713 Hz	188.098833 Hz
223.481103 Hz	223.34894 Hz
237.734345 Hz	237.547289 Hz
245.229167 Hz	245.008992 Hz
250.226527 Hz	249.982589 Hz
254.441903 Hz	254.176854 Hz
258.748788 Hz	258.461147 Hz
263.682892 Hz	263.368079 Hz
269.625224 Hz	269.275824 Hz
276.875267 Hz	276.480859 Hz
285.679291 Hz	285.225972 Hz
296.240074 Hz	295.709751 Hz
308.719867 Hz	308.08933 Hz
323.24202 Hz	322.481799 Hz
339.893522 Hz	338.966522 Hz
358.728968 Hz	357.588881 Hz
379.775347 Hz	378.364823 Hz
403.035042 Hz	401.283648 Hz
428.459824 Hz	426.282331 Hz
445.0769 Hz	442.589095 Hz
456.411232 Hz	453.696729 Hz
486.150759 Hz	482.780906 Hz

In order to try to clarify the influence of the damping as related to the eigenvalue response, a smaller percentage of the critical damping of 1% was chosen. As is seen in Figure 8.3.42 for the real relative error and in Figure 8.3.43 for the imaginary relative error, the response near the eigenvalues is clearly shown. The comparison of Figure 8.3.42 to Figure 8.3.20, the relative error of the case with no damping, as expected, clearly indicates that the peaks near the eigenvalues are reduced in height and more rounded. Hence, at 1% of the critical damping the obvious influence of the eigenvalues is present. The % error for both the real and imaginary magnitude of the displacement of the master DOF is shown in Figures 8.3.44 and 8.3.45. It is interesting to note that the % error of ≈0.01% is nearly identical to that of the undamped case

(Figure 8.3.21). Also, as in the undamped case (Figure 8.3.22), there is no discernable difference between the reduced response and the full solution response in Figures 8.3.46 and 8.3.47. The further reduction in the percentage of the critical damping of 0.5% was examined. Figures 8.3.48 to 8.3.53 show identical results to the previous damping cases. The peak heights and peak rounding are less than the undamped case but slightly more than the 1% damping case.

The comparison of the most stringent error criteria, the relative error, for the case in which there is no damping to the cases with different percentages of critical damping demonstrates the effect that damping has on the overall response of this model particularly the response near the eigenvalues. The relative error for the undamped case is compared to that of the three damped cases (real component only) for the Lagrangian (Figure 8.3.54) and the Hermitian interpolation methods (Figure 8.3.55). These figures show clearly the gradual decrease and smoothing if the peaks at (near) the eigenvalues. Also the comparison of the % error of the magnitude of the displacements of the "master" DOF u_{51x} for the undamped and damped cases, Figures 8.3.21, 8.3.28, 8.3.44 and 8.3.50, shows that the overall error is the same at 0.01%. It appears that the accuracy of the FWR method is related to the % DOF reduction, the interpolation window size, the sweep step size and interpolation method used and not whether damping is present. This could allow analyses to be performed with out damping in order to determine the important parameters for the analysis and then used to examine cases with damping.

It is recognized that perhaps more appropriate choices of the frequency modes used to calculate the Rayleigh damping for this model could be made. Initially the frequency modes chosen were approximately in the middle of the interpolation window. These were selected not necessarily to represent a real problem with damping but rather these were preliminary choices used to determine the applicability of the FWR method to systems with Rayleigh damping.

Spherical Half-Shell Model Figures

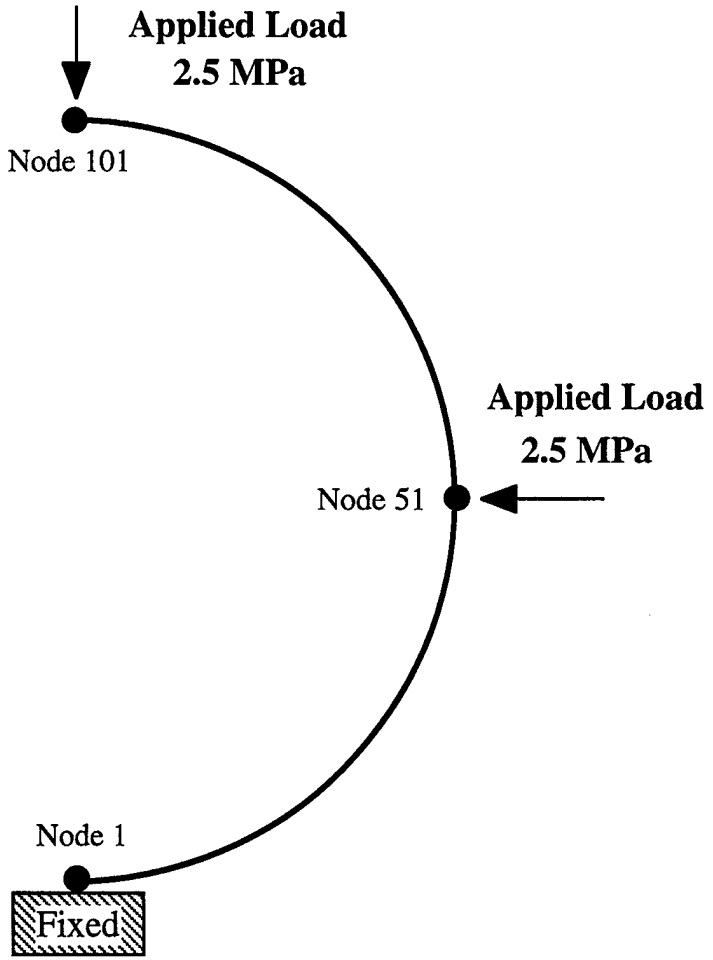


Figure 8.3.1 – Half-Spherical Shell Model

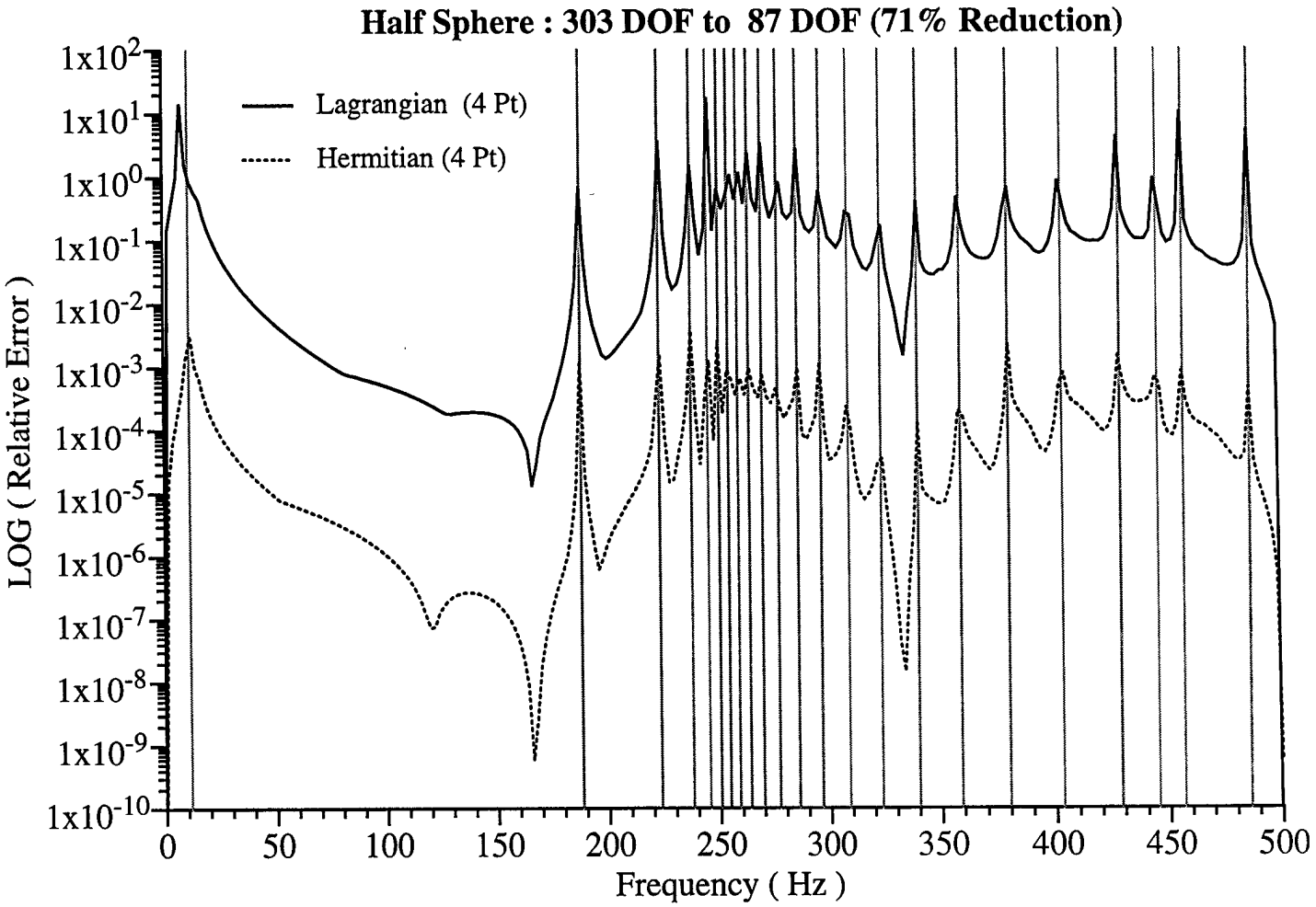


Figure 8.3.2 – Relative Error comparing interpolations for the frequency range of 0 to 500 Hz and a frequency step size of 2 Hz.

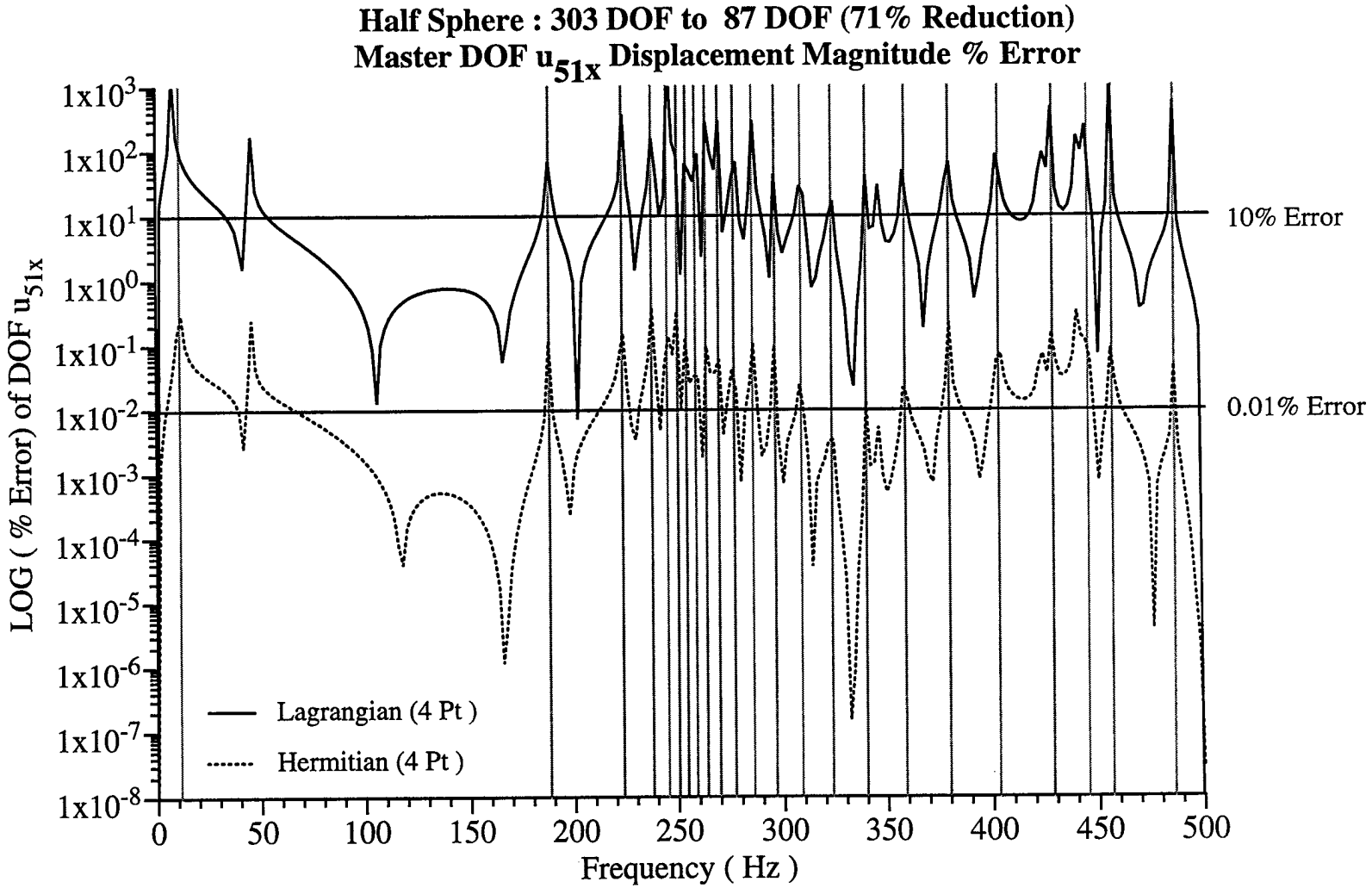


Figure 8.3.3 – % Error for the "master" DOF $|u|_{51x}$ comparing interpolations for the range of 0 to 500 Hz and a step size of 2 Hz.

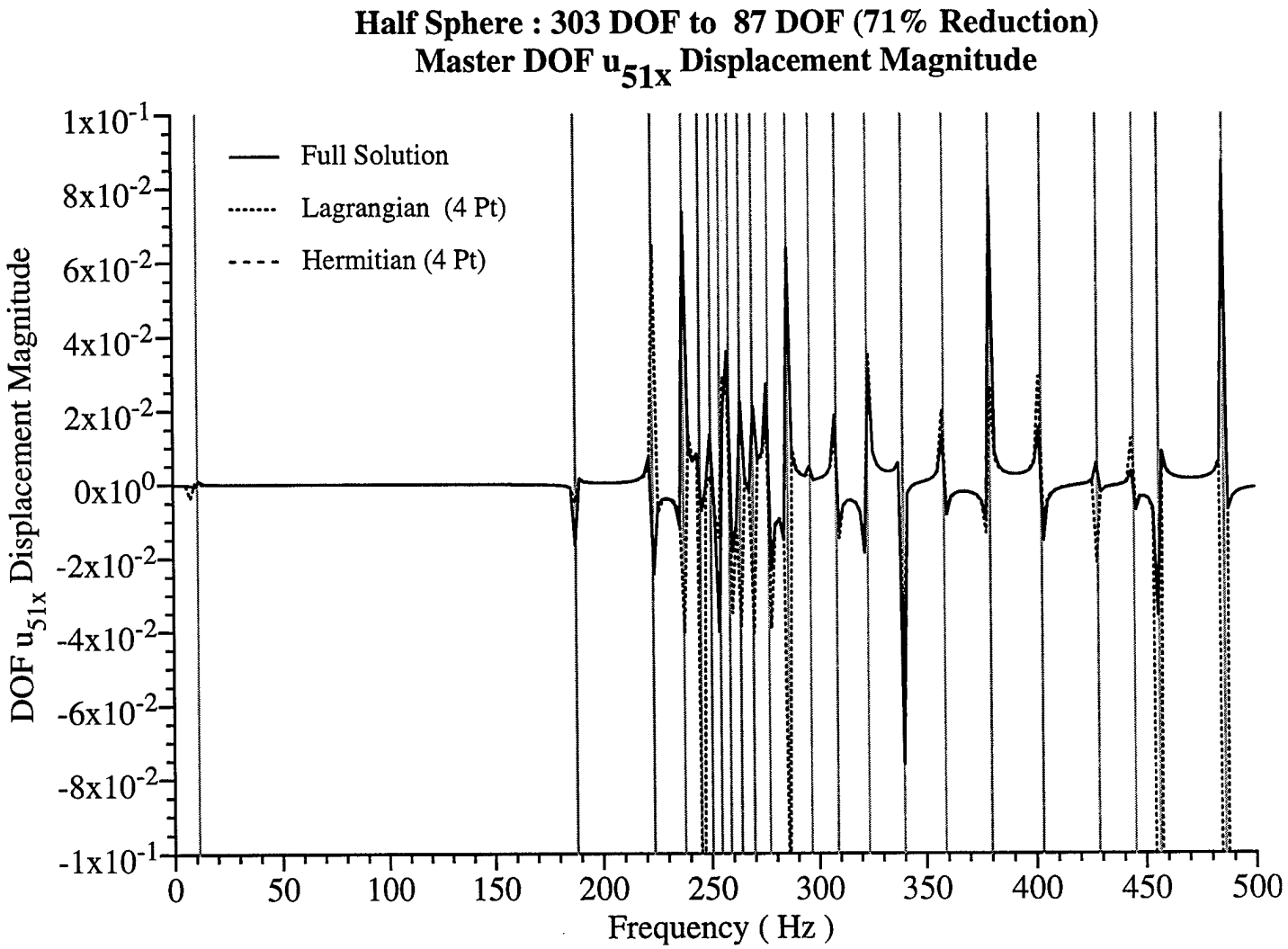


Figure 8.3.4 – The "master" DOF $|u|_{51x}$ (magnitude of the displacement) comparing interpolation functions for the range of 0 to 500 Hz and a step size of 2 Hz.

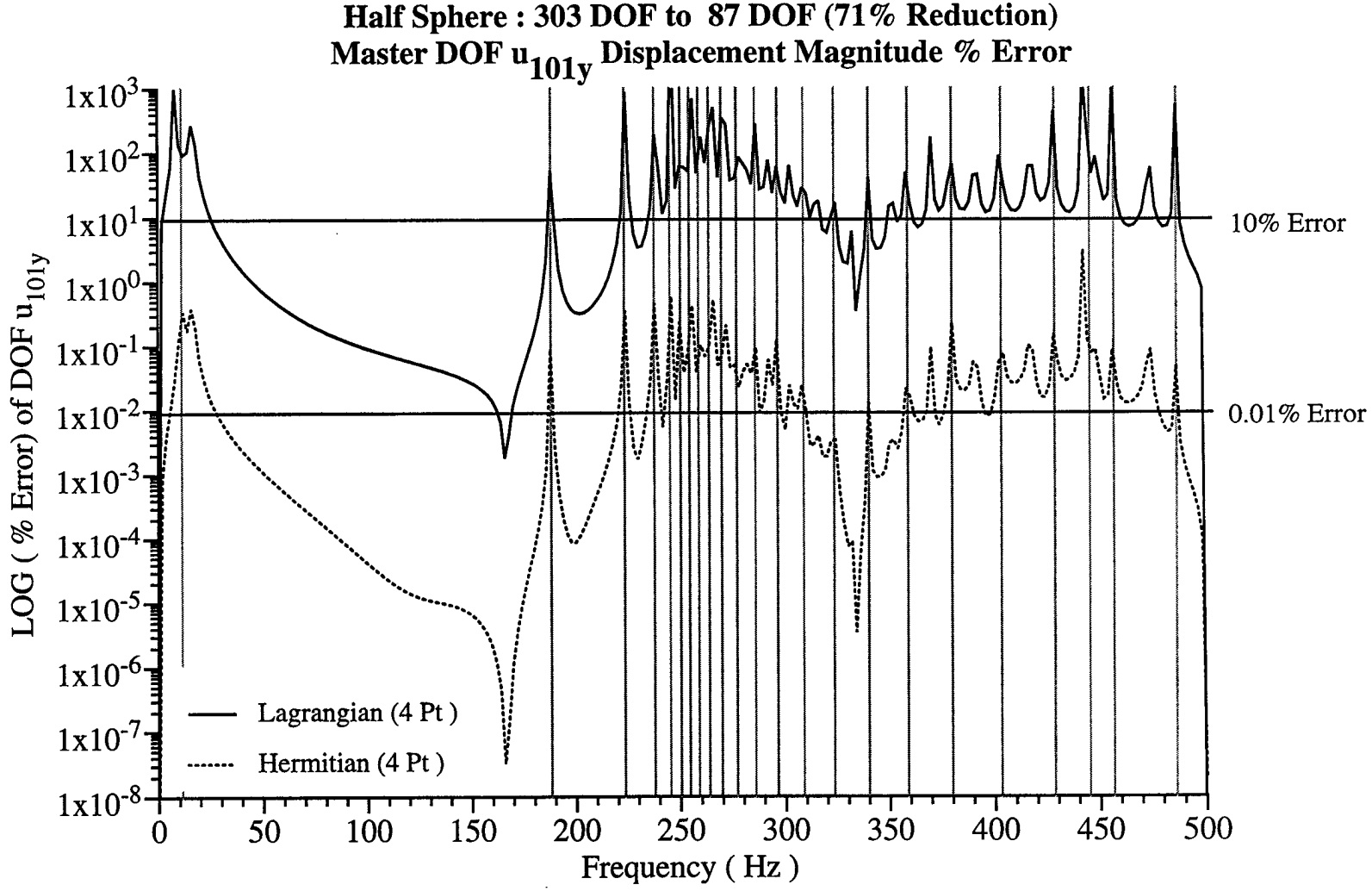


Figure 8.3.5 – % Error for the "master" DOF $|u|_{101y}$ comparing interpolations for the range of 0 to 500 Hz and a frequency step size of 2 Hz.

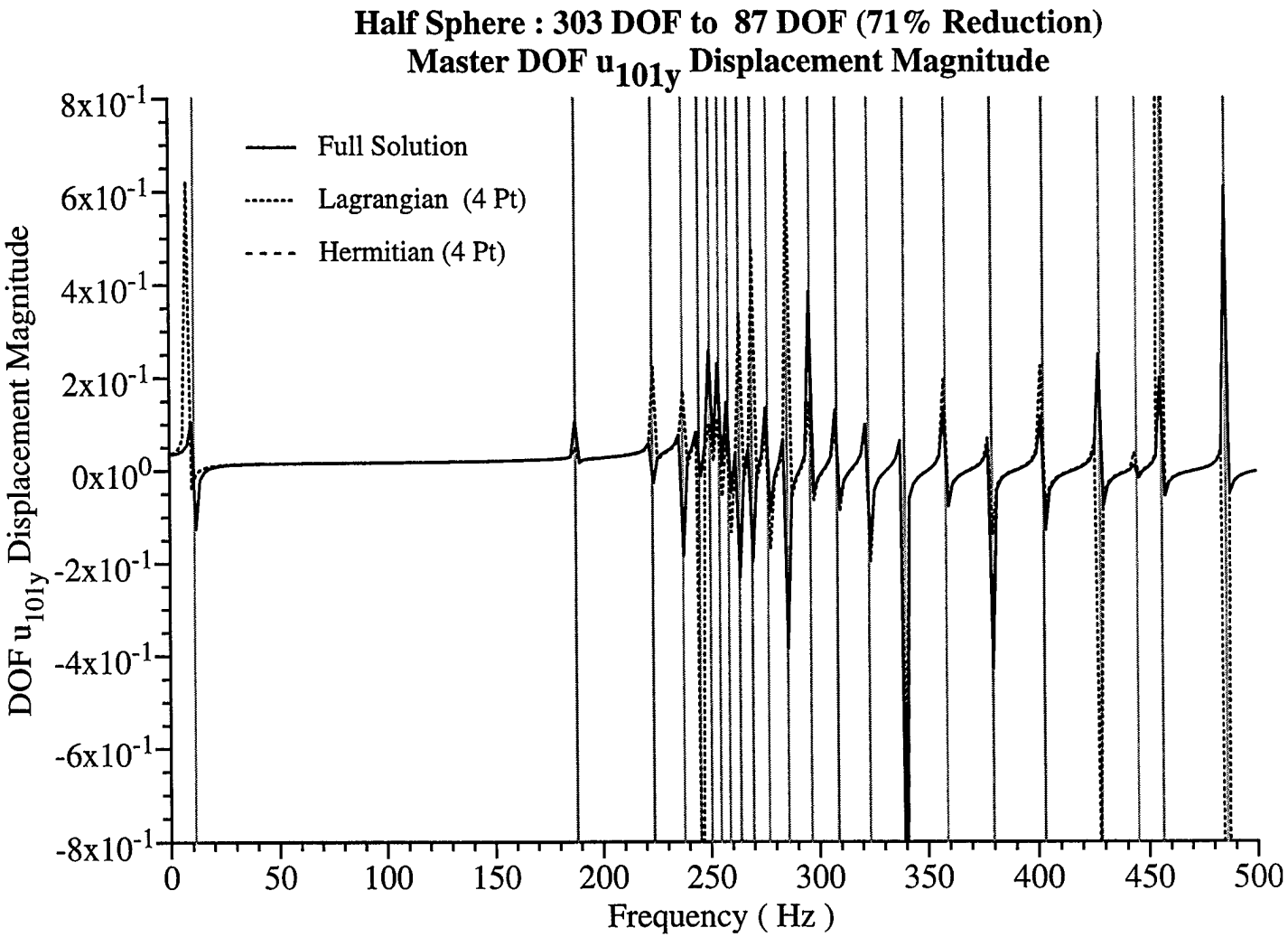


Figure 8.3.6 – The "master" DOF $|u|_{101y}$ (magnitude of the displacement) comparing interpolation functions for the frequency range of 0 to 500 Hz and a frequency step size of 2 Hz.

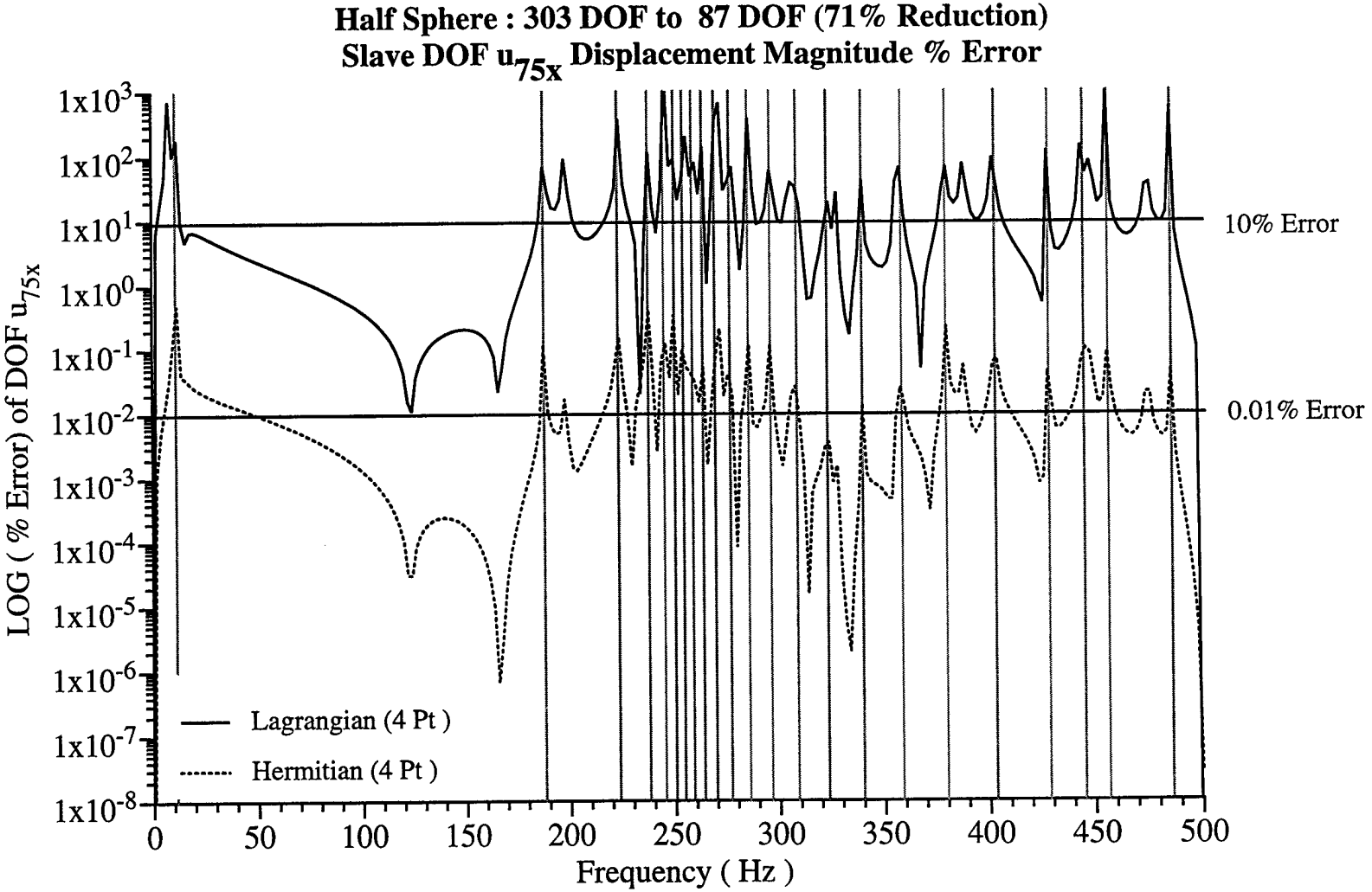


Figure 8.3.7 – % Error for the reconstructed "slave" DOF $|u|_{75x}$ comparing interpolations for the range of 0 to 500 Hz and a step size of 2 Hz.

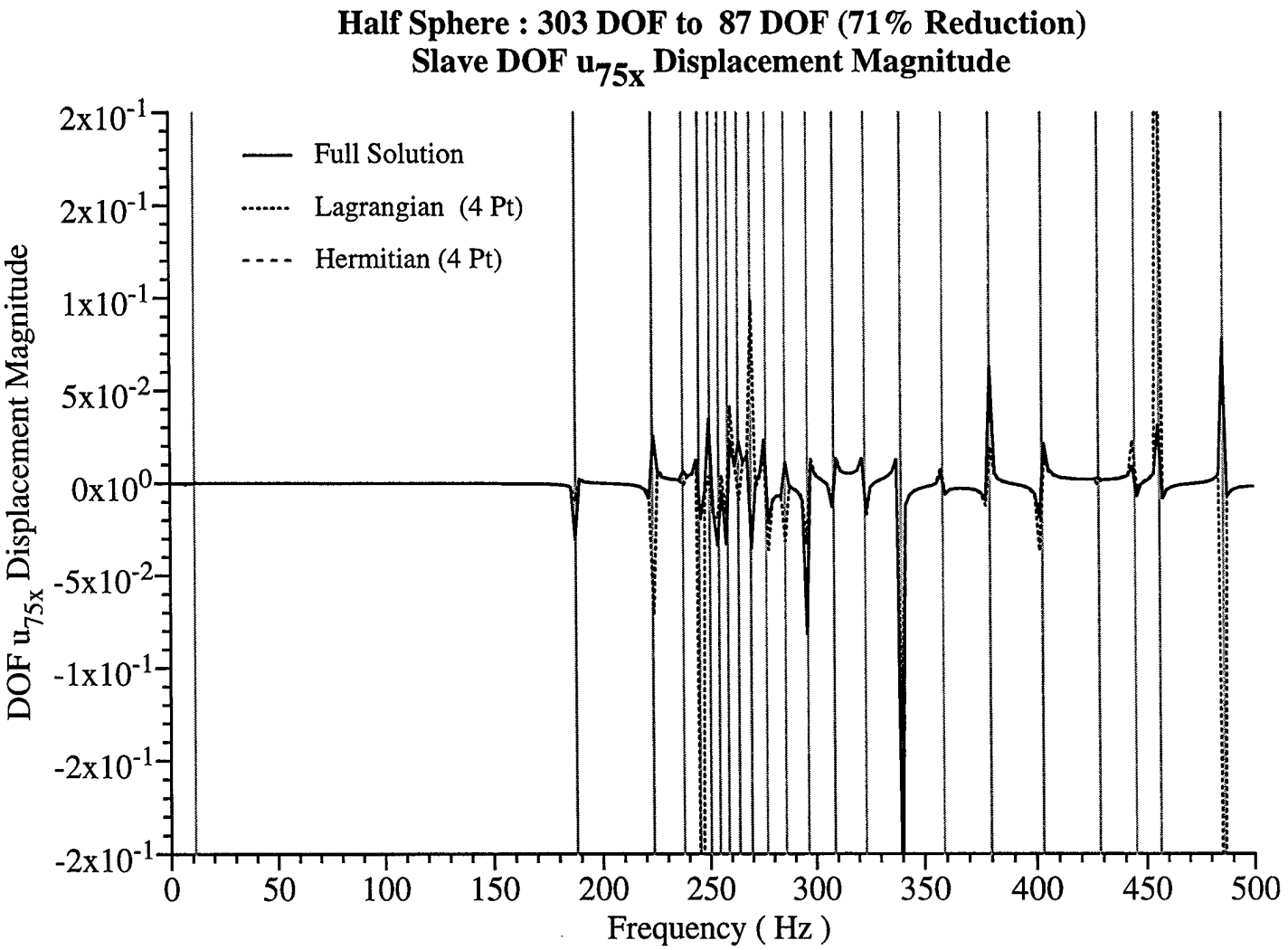


Figure 8.3.8 – The reconstructed "slave" DOF lul_{75x} for the frequency range of 0 to 500 Hz.

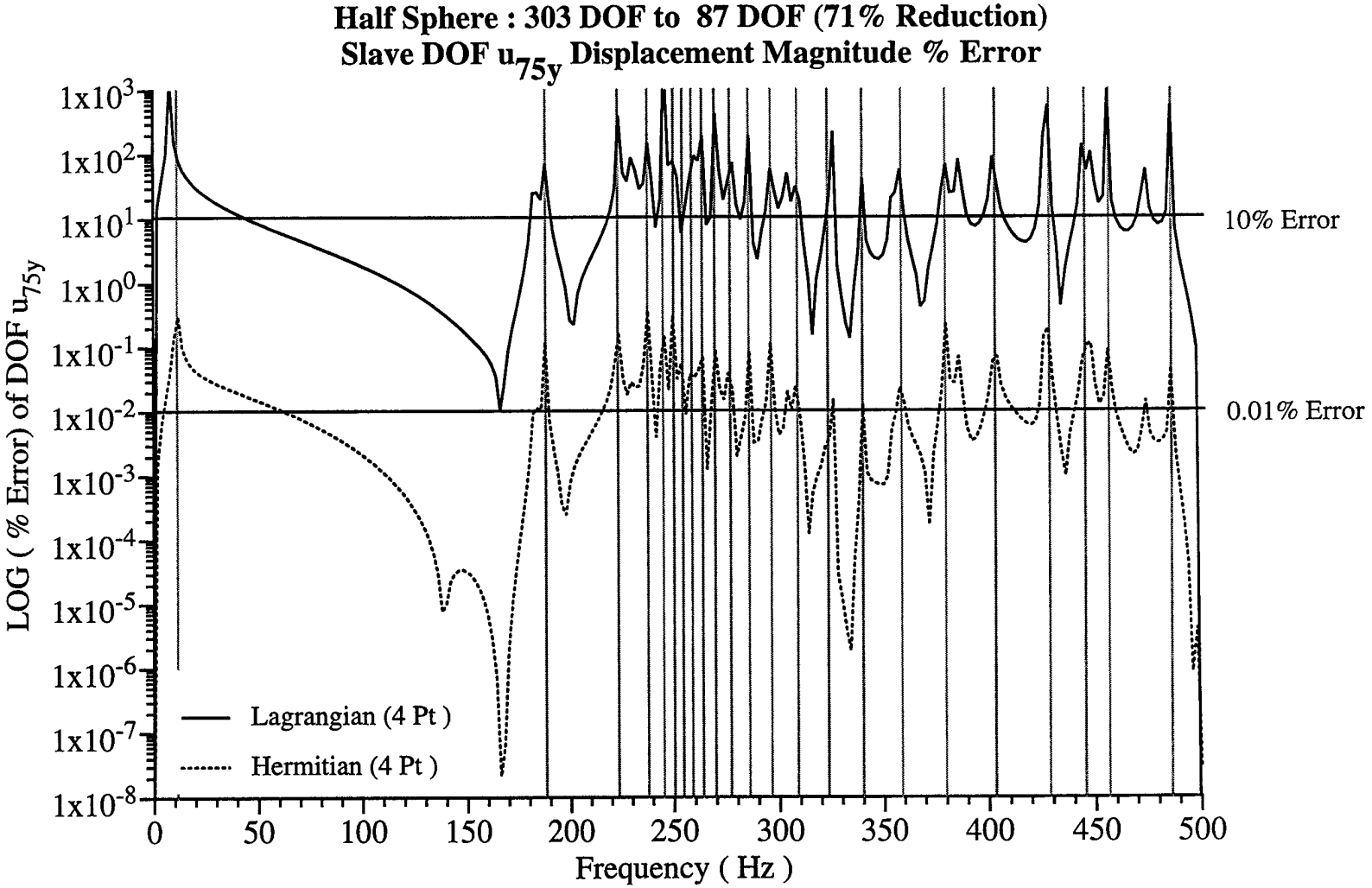


Figure 8.3.9 – % Error for the reconstructed "slave" DOF $|u|_{75y}$ for the range of 0 to 500 Hz and a step of 2 Hz.

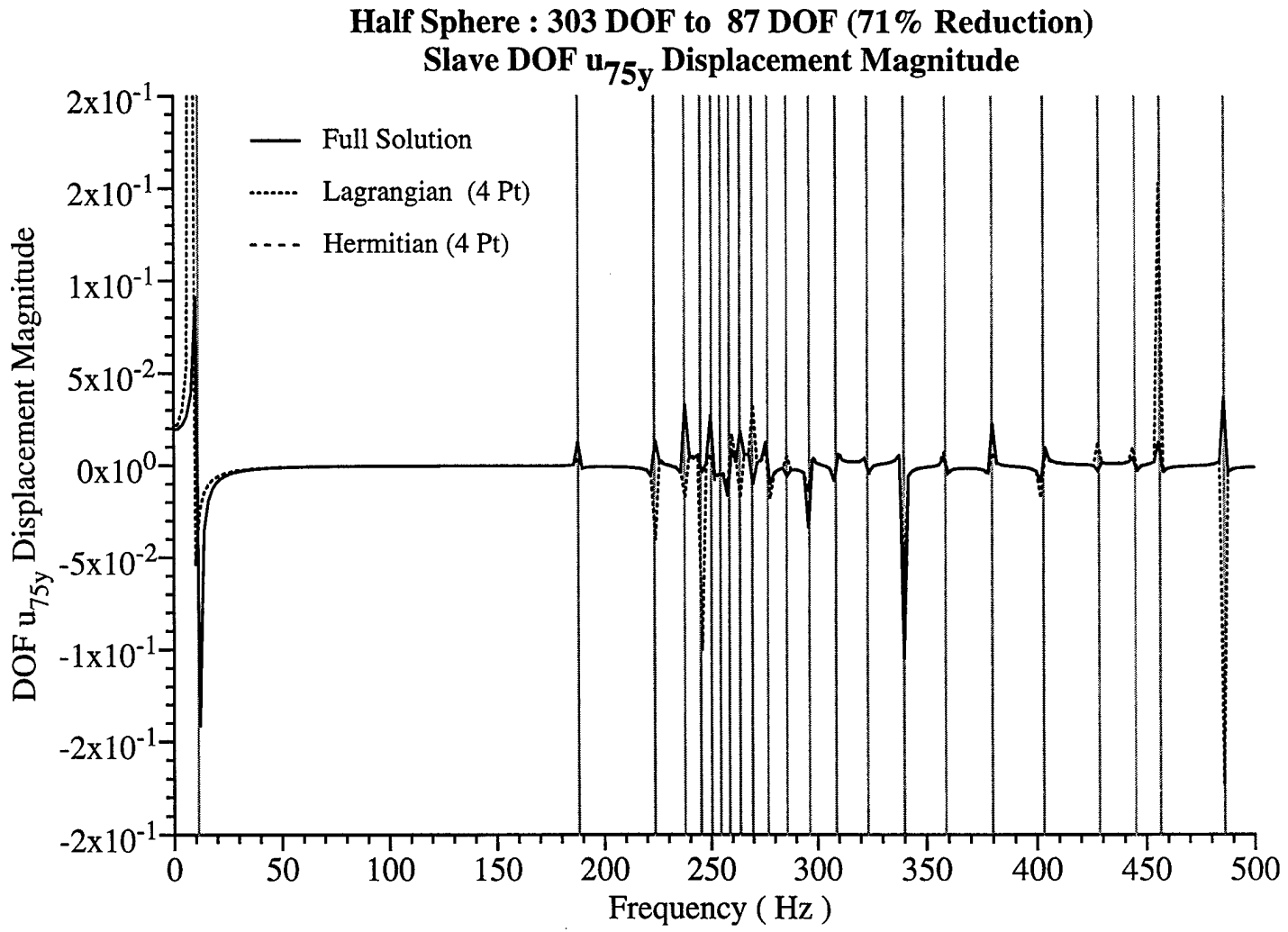


Figure 8.3.10 – The reconstructed "slave" DOF $|u|_{75y}$ for the frequency range of 0 to 500 Hz.

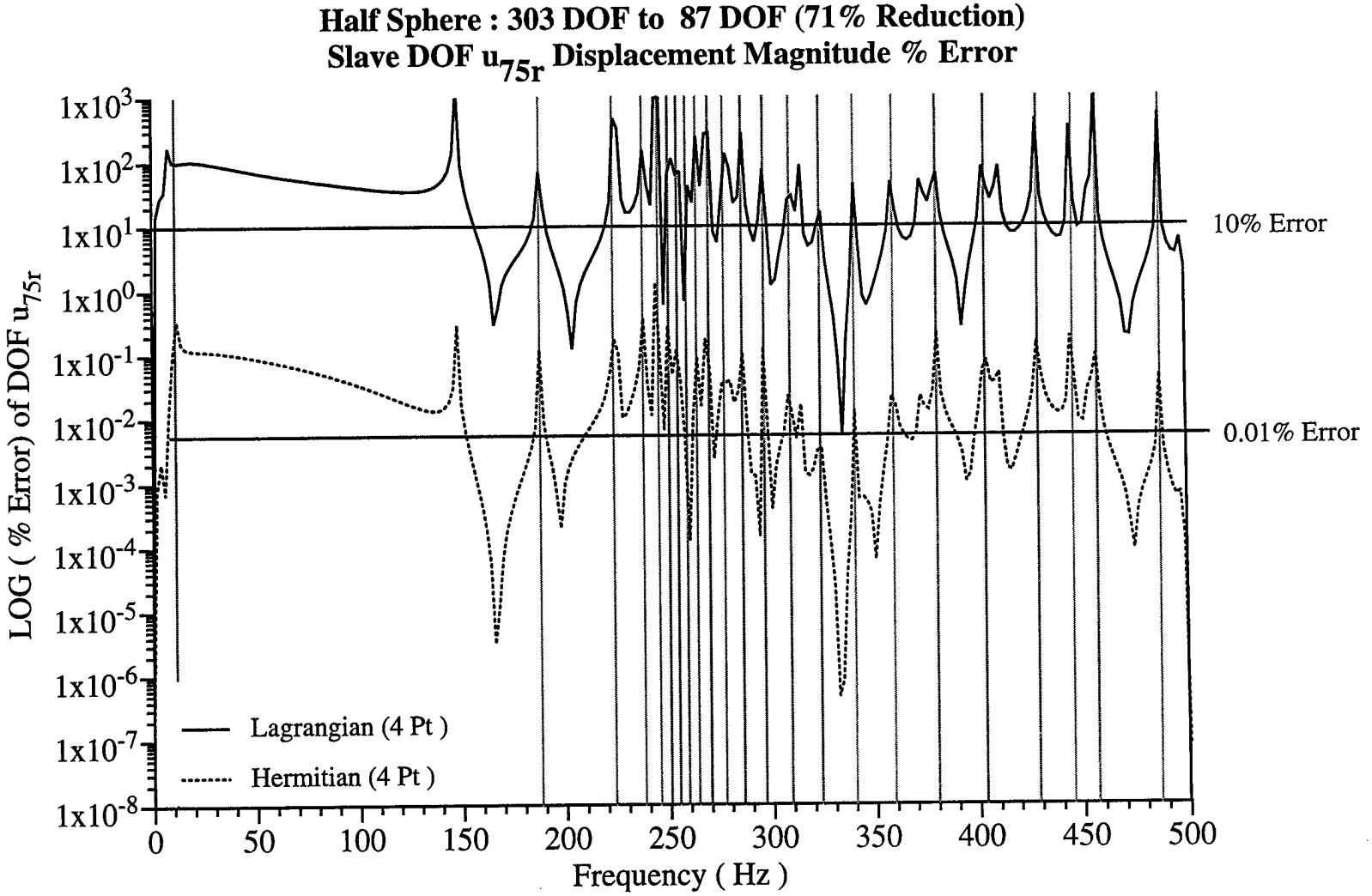


Figure 8.3.11 – % Error for the reconstructed "slave" DOF lul_{7rx} for the frequency range of 0 to 500 Hz.

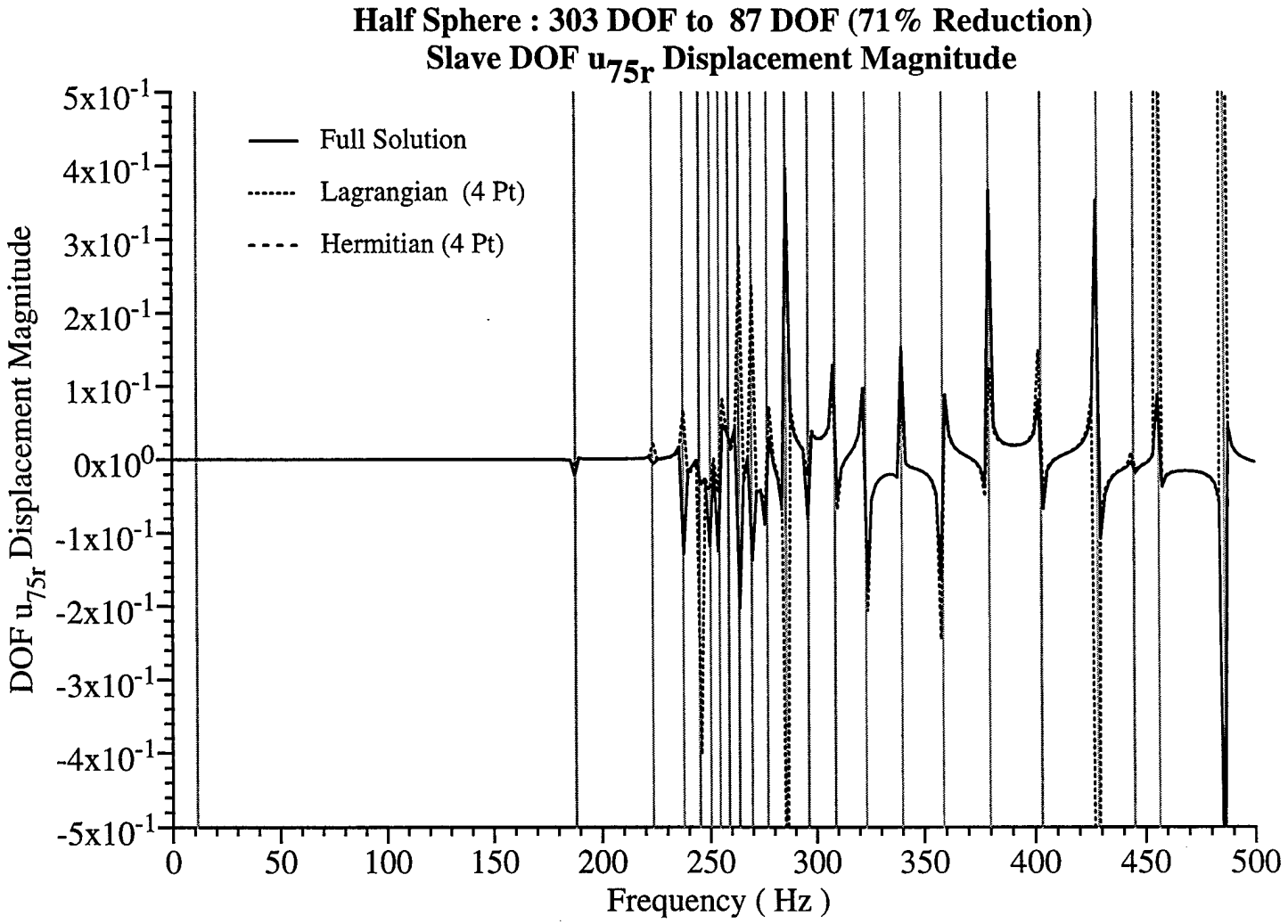


Figure 8.3.12 – The reconstructed "slave" DOF $|u|_{75r}$ for the frequency range of 0 to 500 Hz.

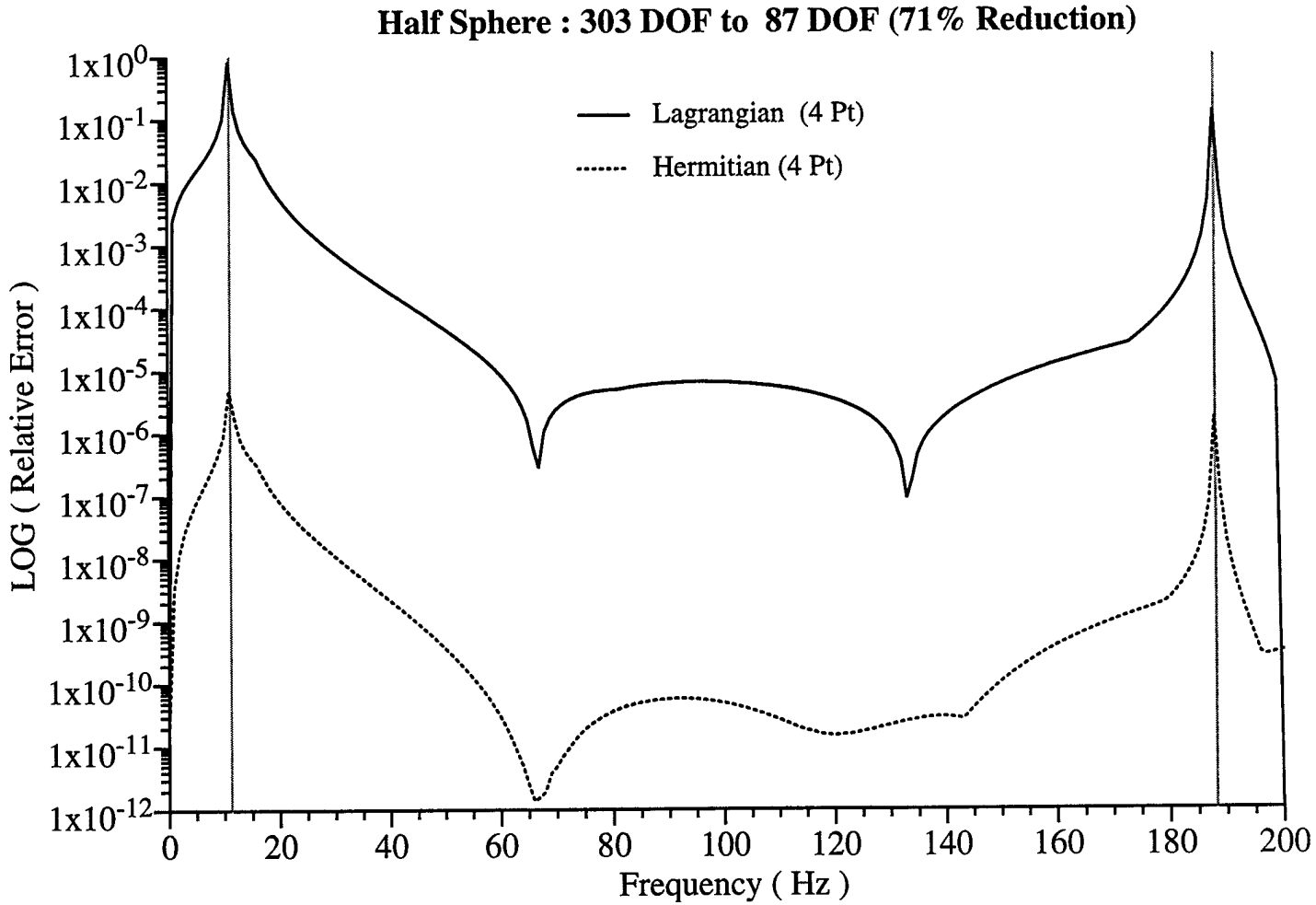


Figure 8.3.13 – Relative Error comparing interpolations for the range of 0 to 200 Hz and a step size of 1 Hz.

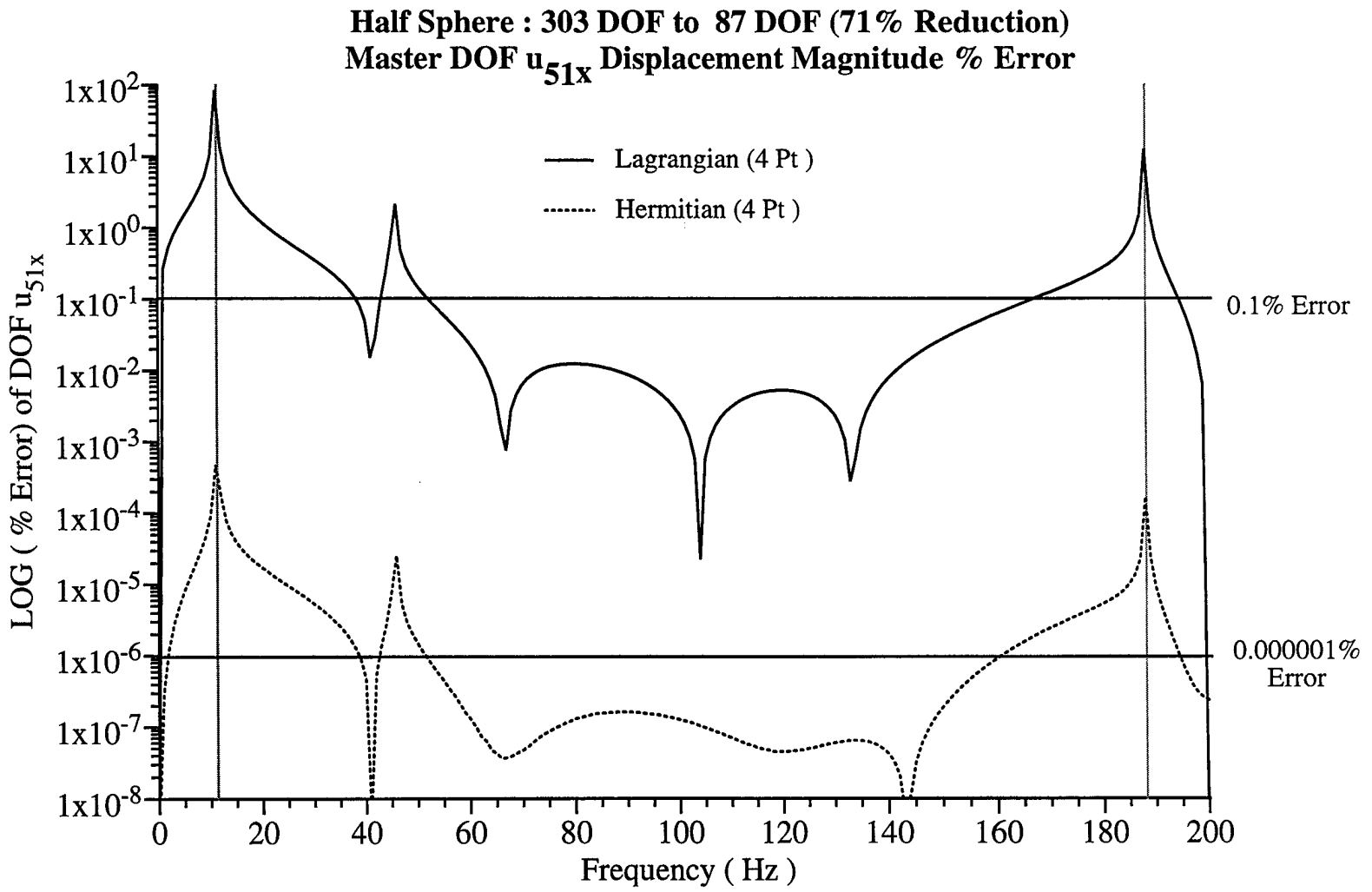


Figure 8.3.14 – % Error for the "master" DOF $|u|_{51x}$ comparing interpolations for the range of 0 to 200 Hz and a step size of 1 Hz. NOTE: Hermitian % error.

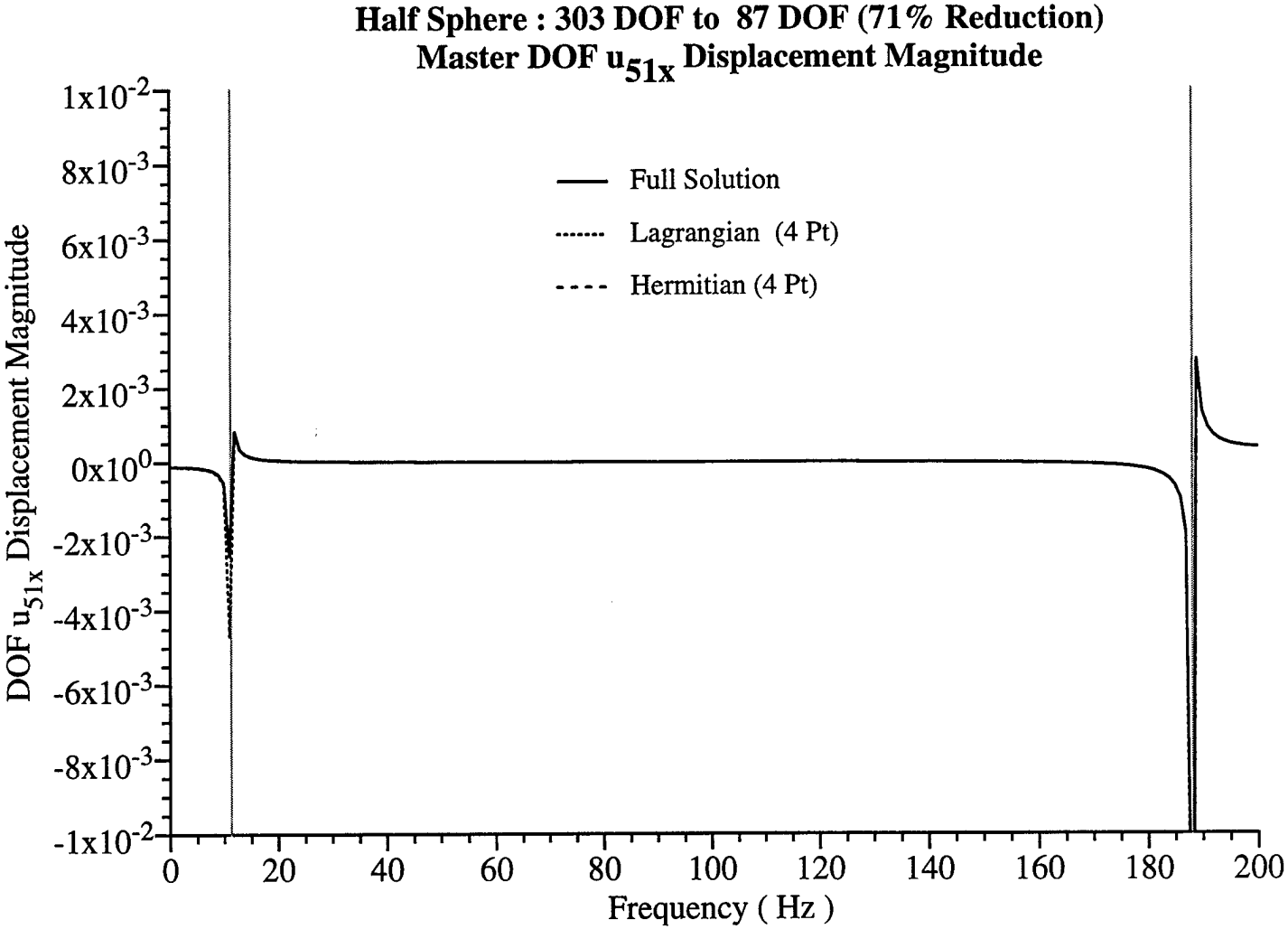


Figure 8.3.15 – The "master" DOF $|u|_{51x}$ (magnitude of the displacement) comparing interpolation functions for the range of 0 to 200 Hz.

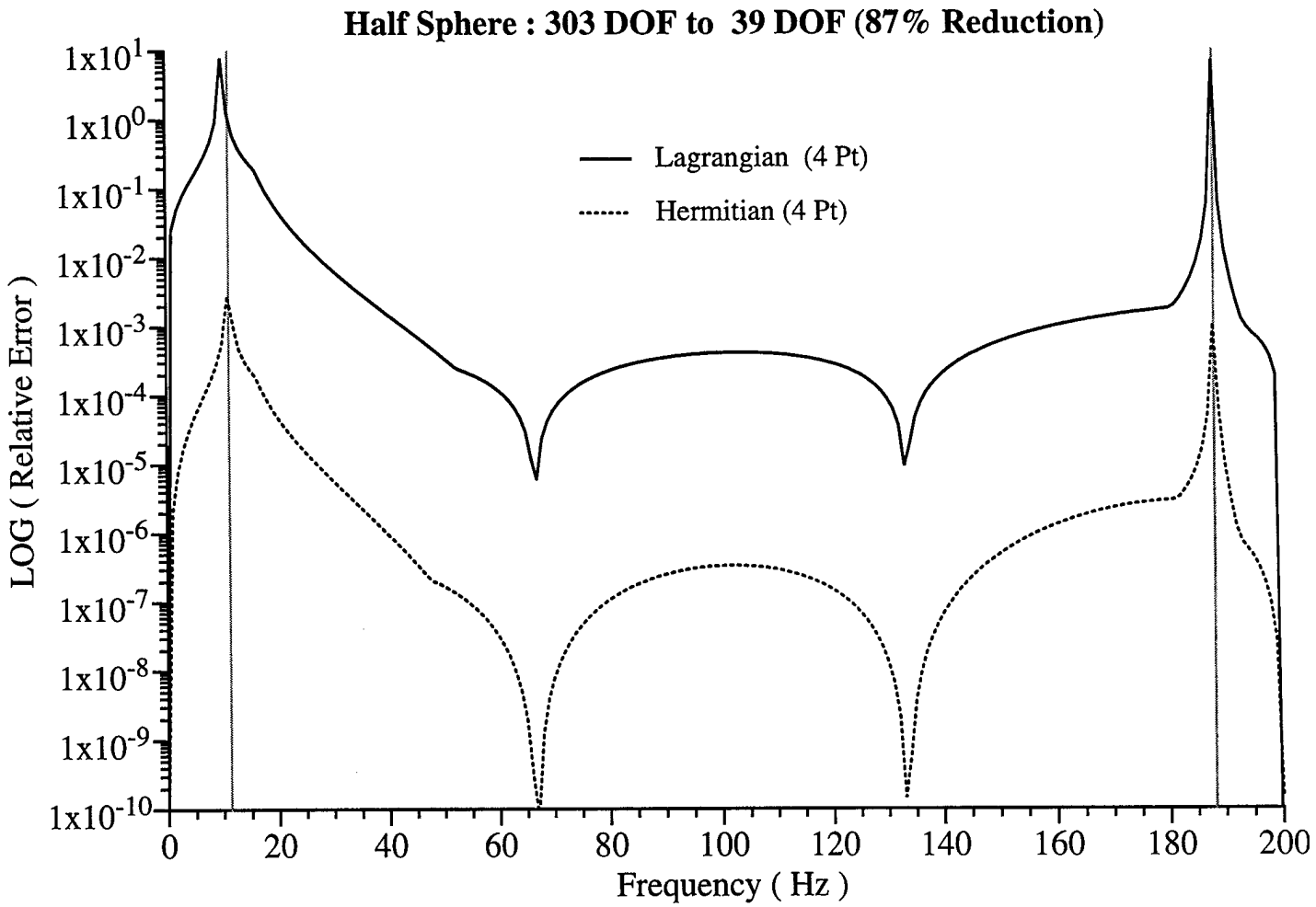


Figure 8.3.16 – Relative Error comparing interpolation functions for the frequency range of 0 to 200 Hz and a frequency step size of 1 Hz. NOTE: 87% DOF Reduction.

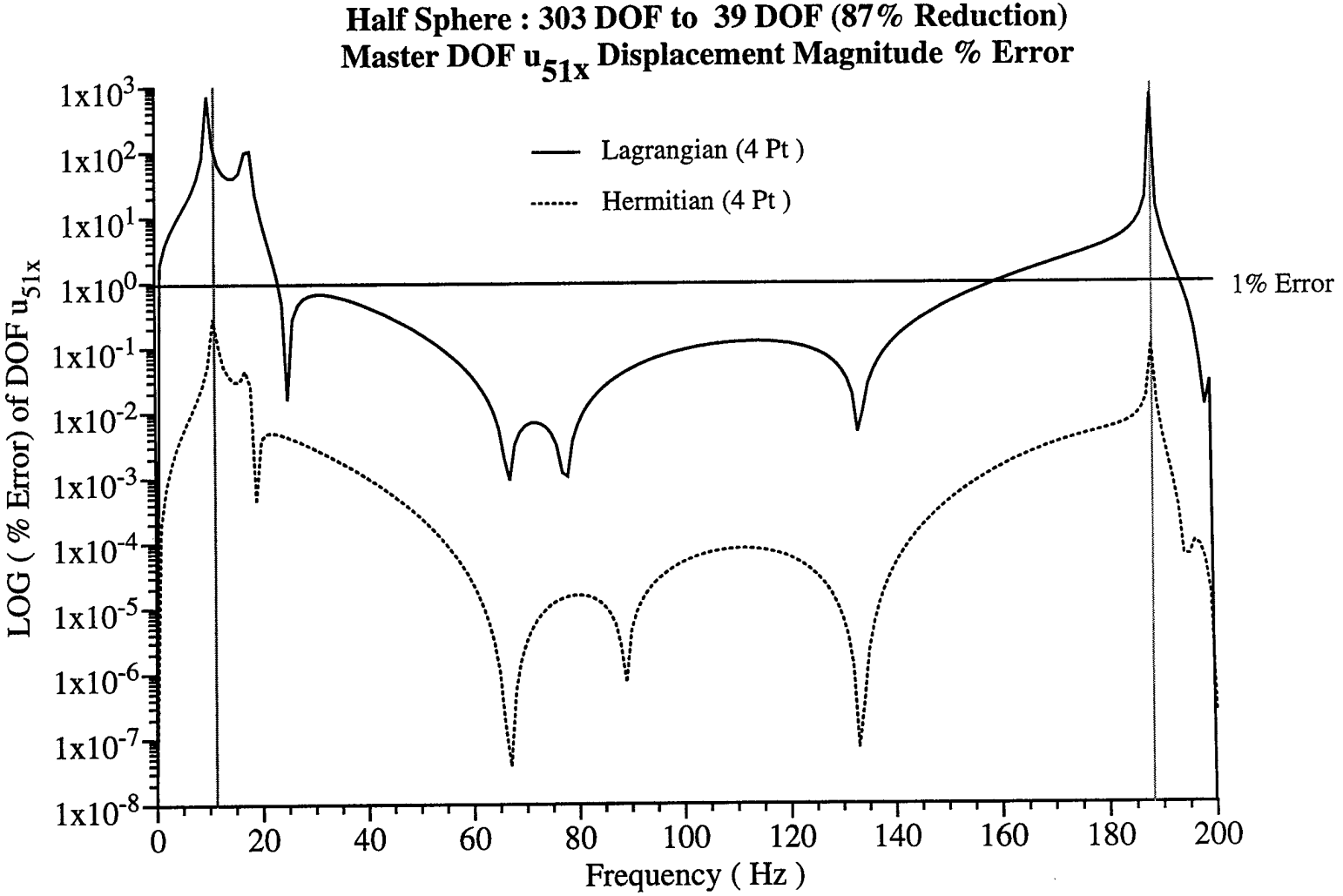


Figure 8.3.17 – % Error for the "master" DOF $|u|_{51x}$ comparing interpolation functions for the frequency range of 0 to 200 Hz. NOTE: 87% DOF Reduction with less than 1% Error.

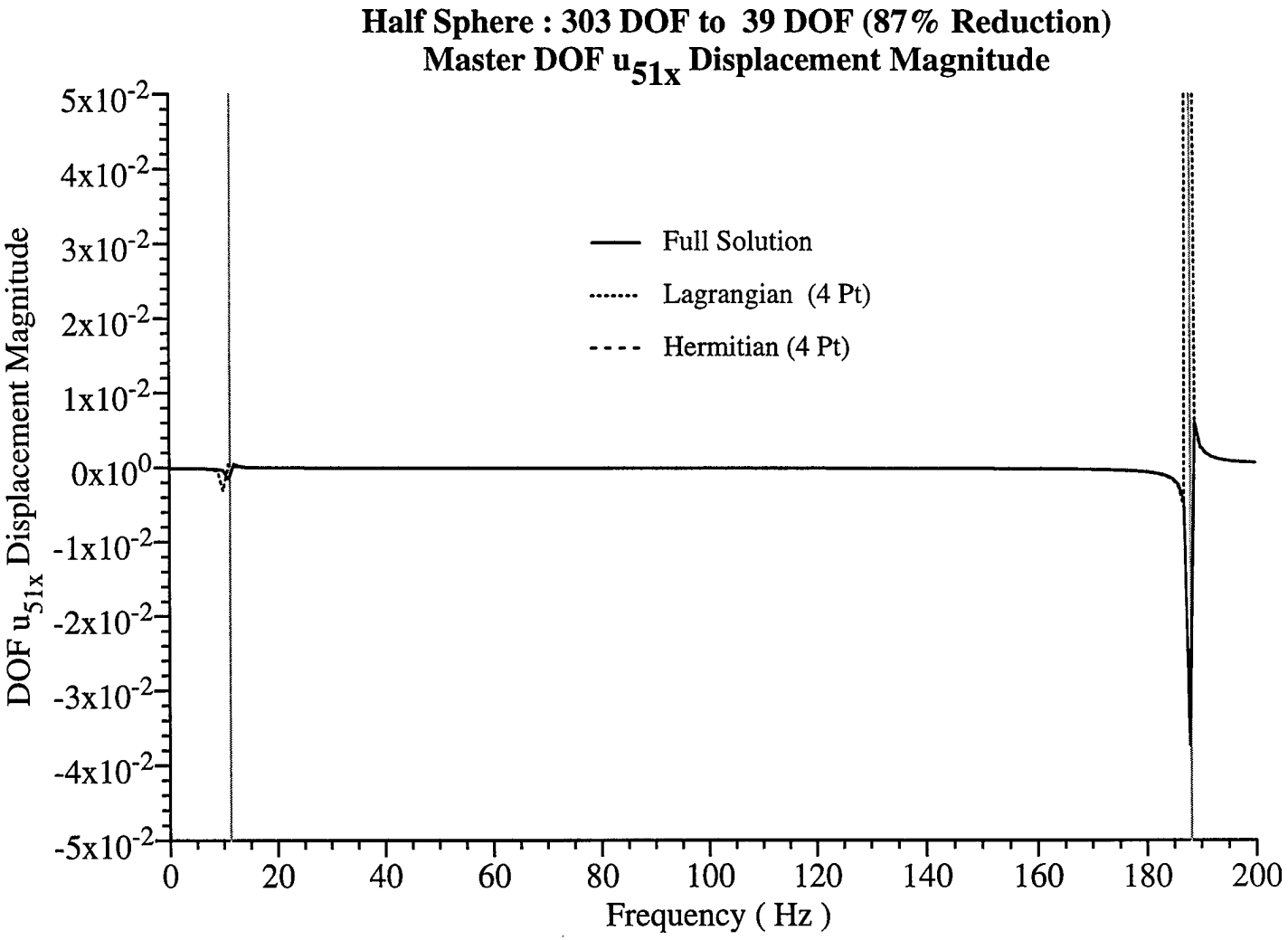


Figure 8.3.18 – The "master" DOF $|u|_{51x}$ for the frequency range of 0 to 500 Hz. NOTE: 87% DOF Reduction.

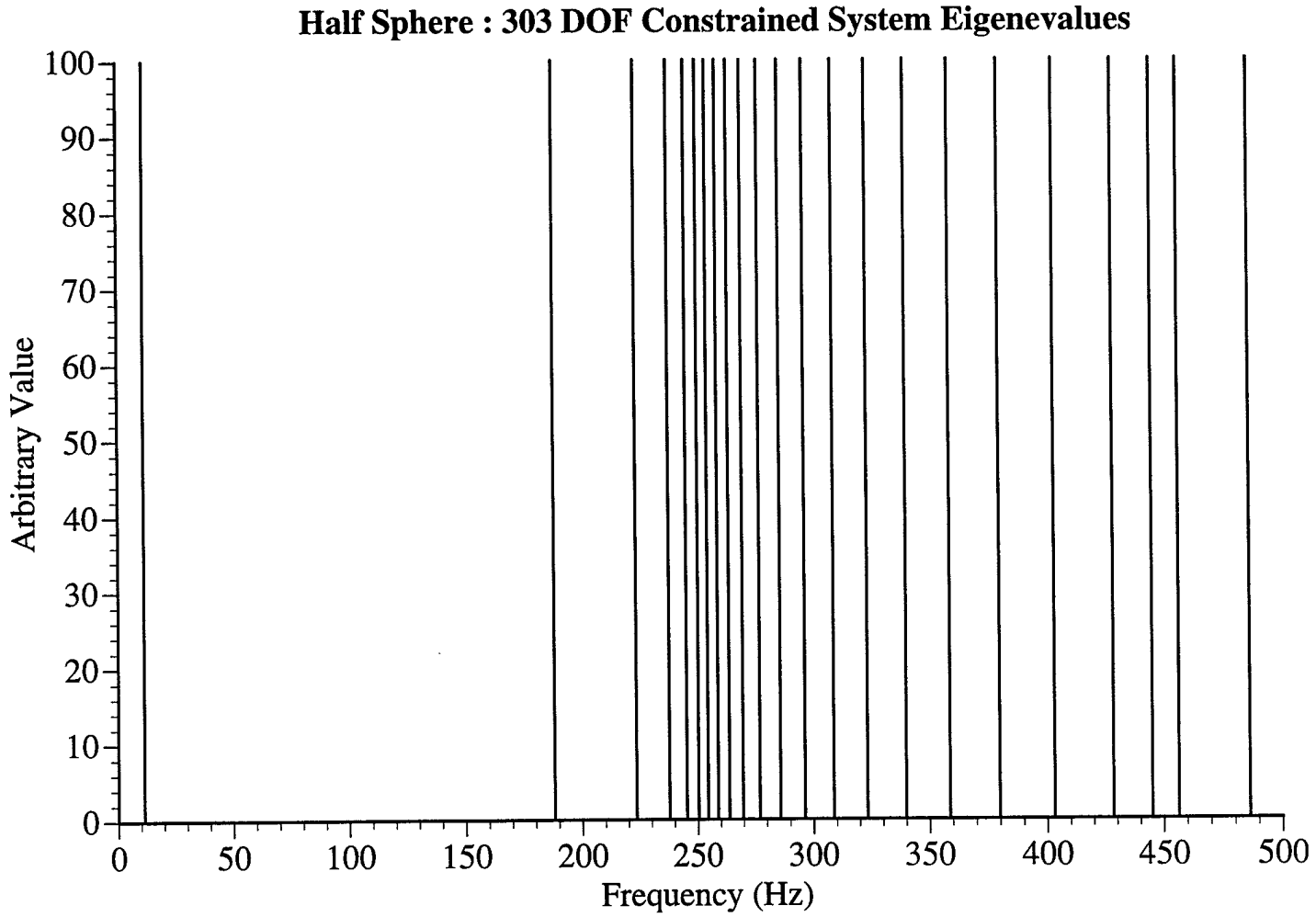


Figure 8.3.19 – The constrained eigenvalues for the spherical half shell model.

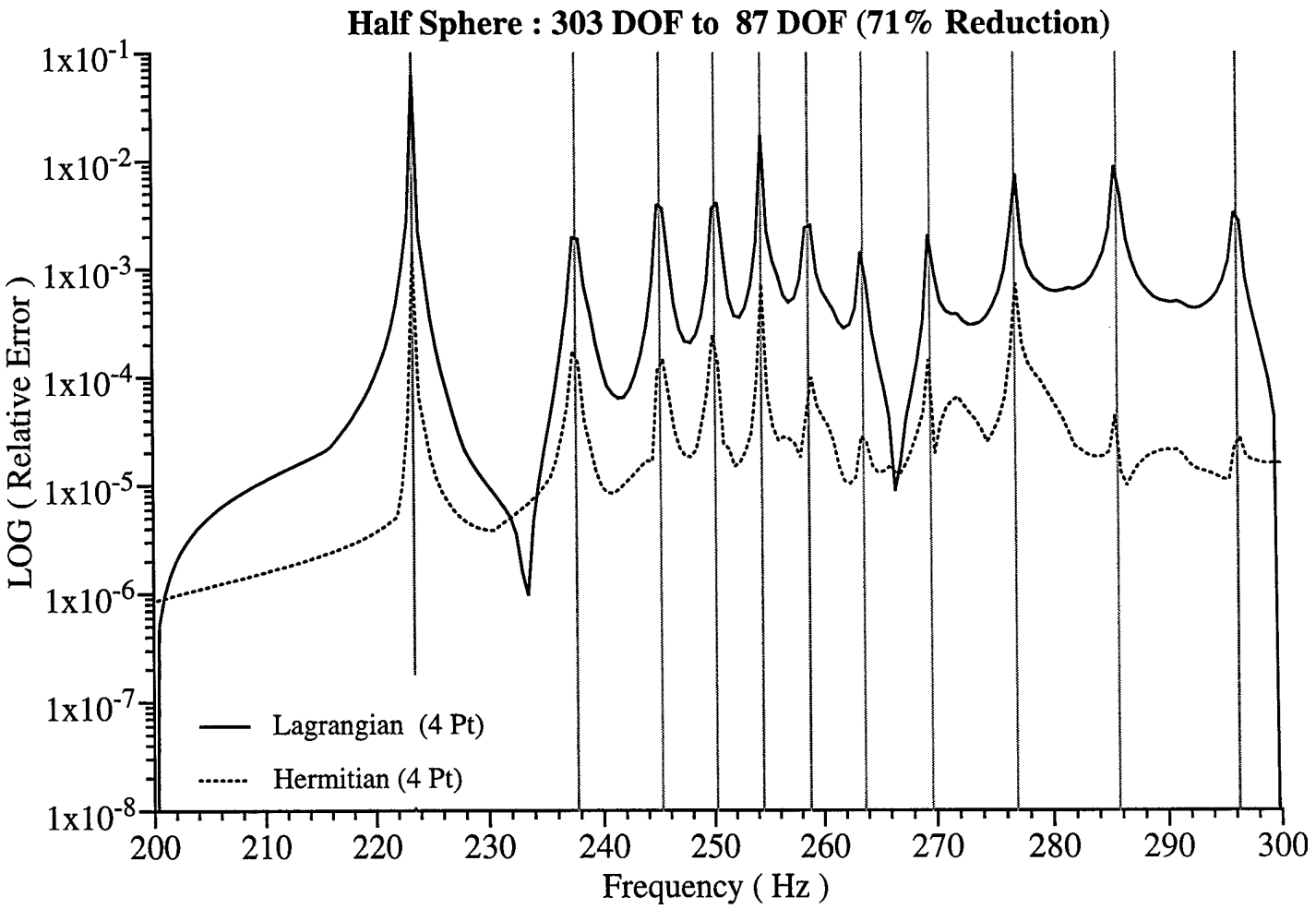


Figure 8.3.20 – Relative Error comparing interpolation functions for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz.

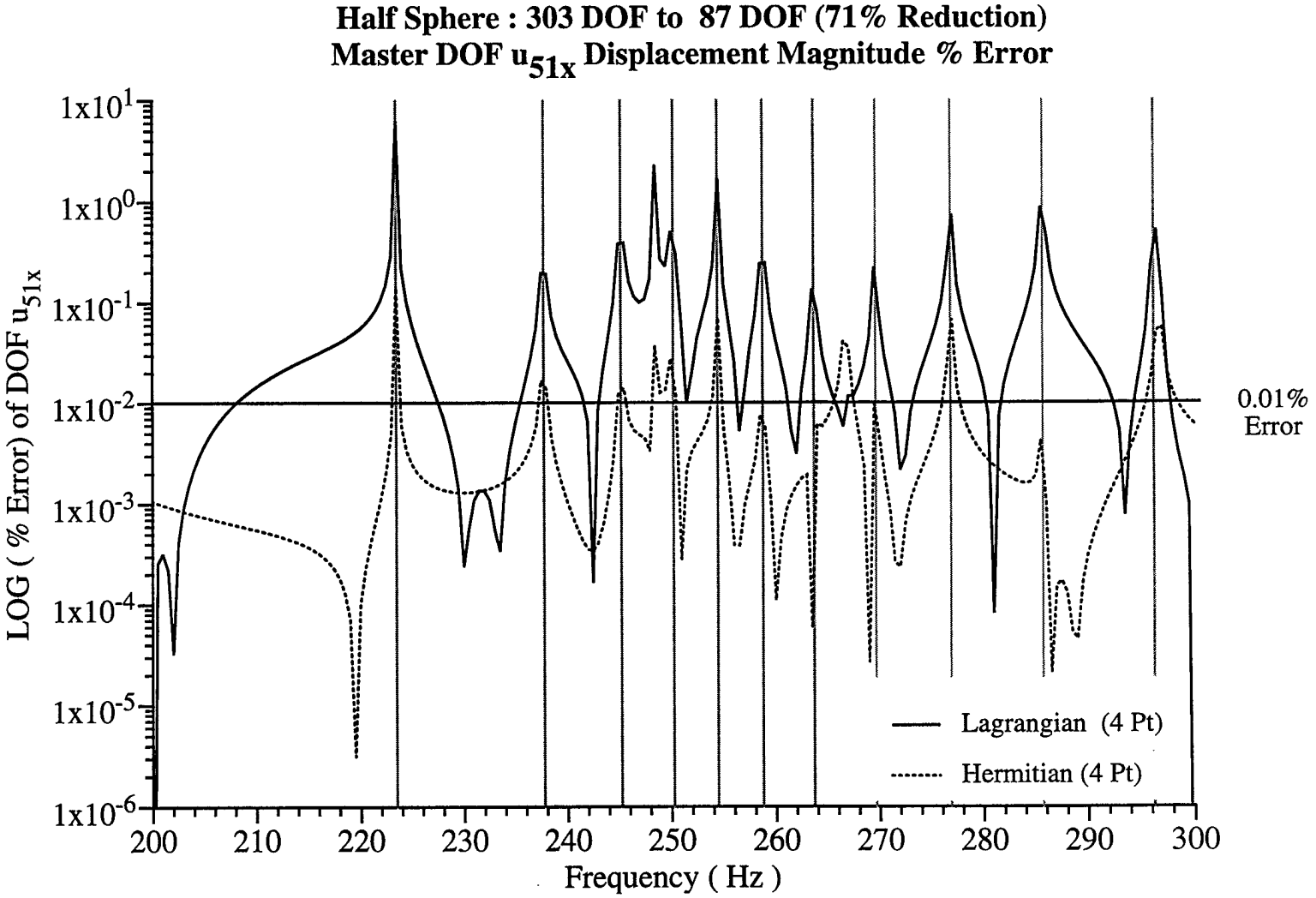


Figure 8.3.21 – % Error for the "master" DOF $|u|_{51x}$ comparing interpolation functions for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz.

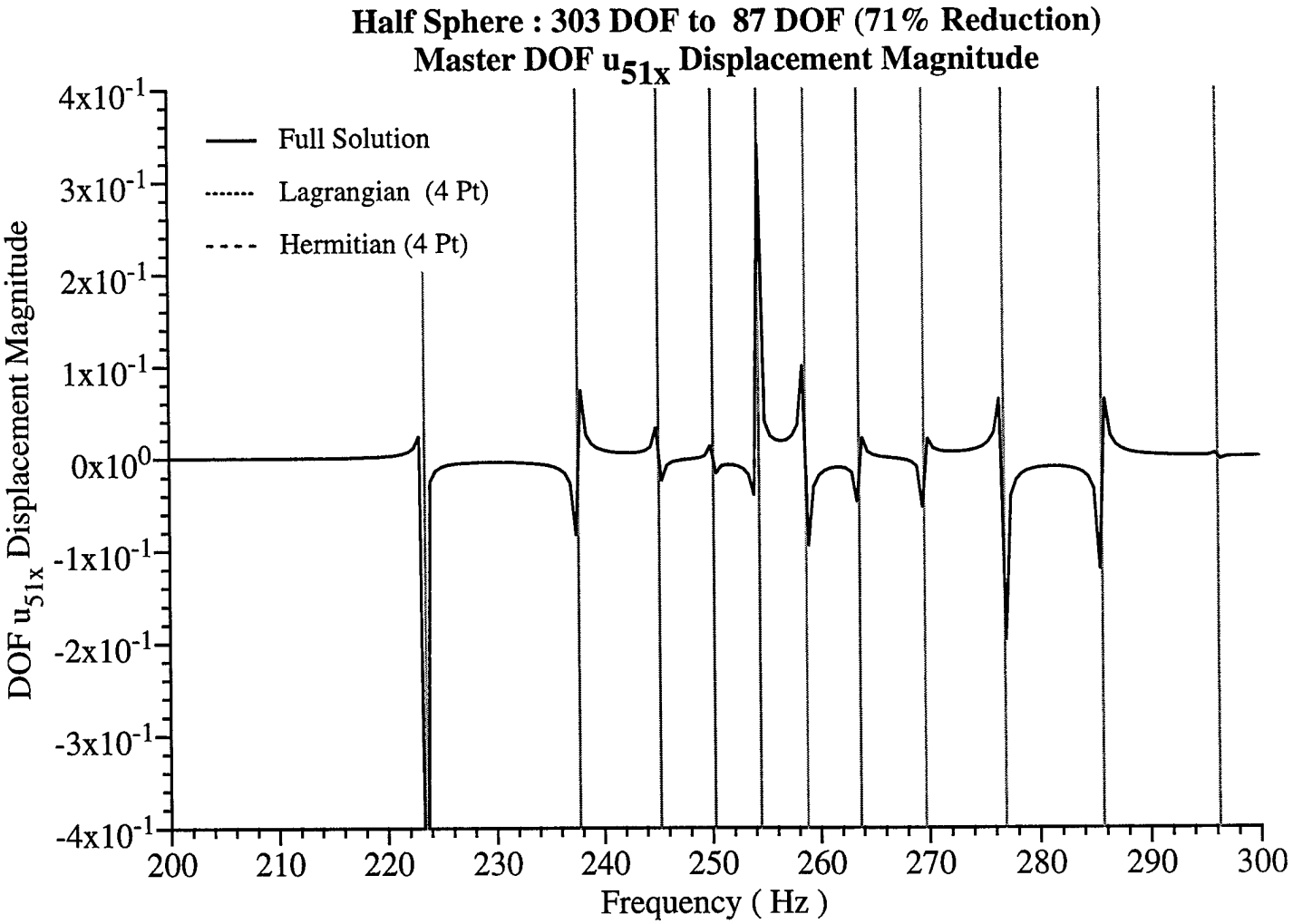


Figure 8.3.22 – The "master" DOF $|u|_{51x}$ comparing interpolation functions for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz.

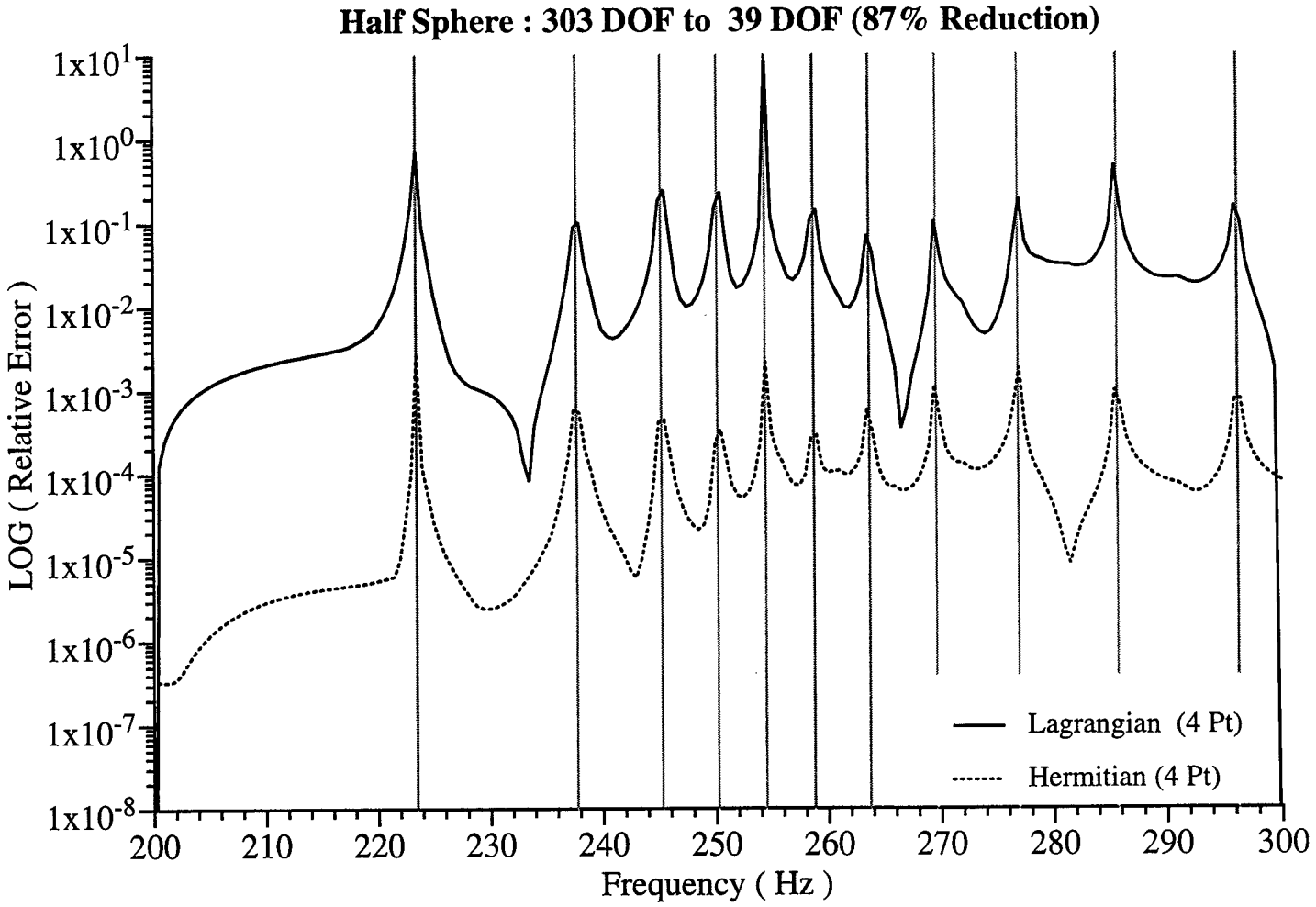


Figure 8.3.23 – Relative Error comparing interpolation functions for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz. NOTE: 87% DOF Reduction.

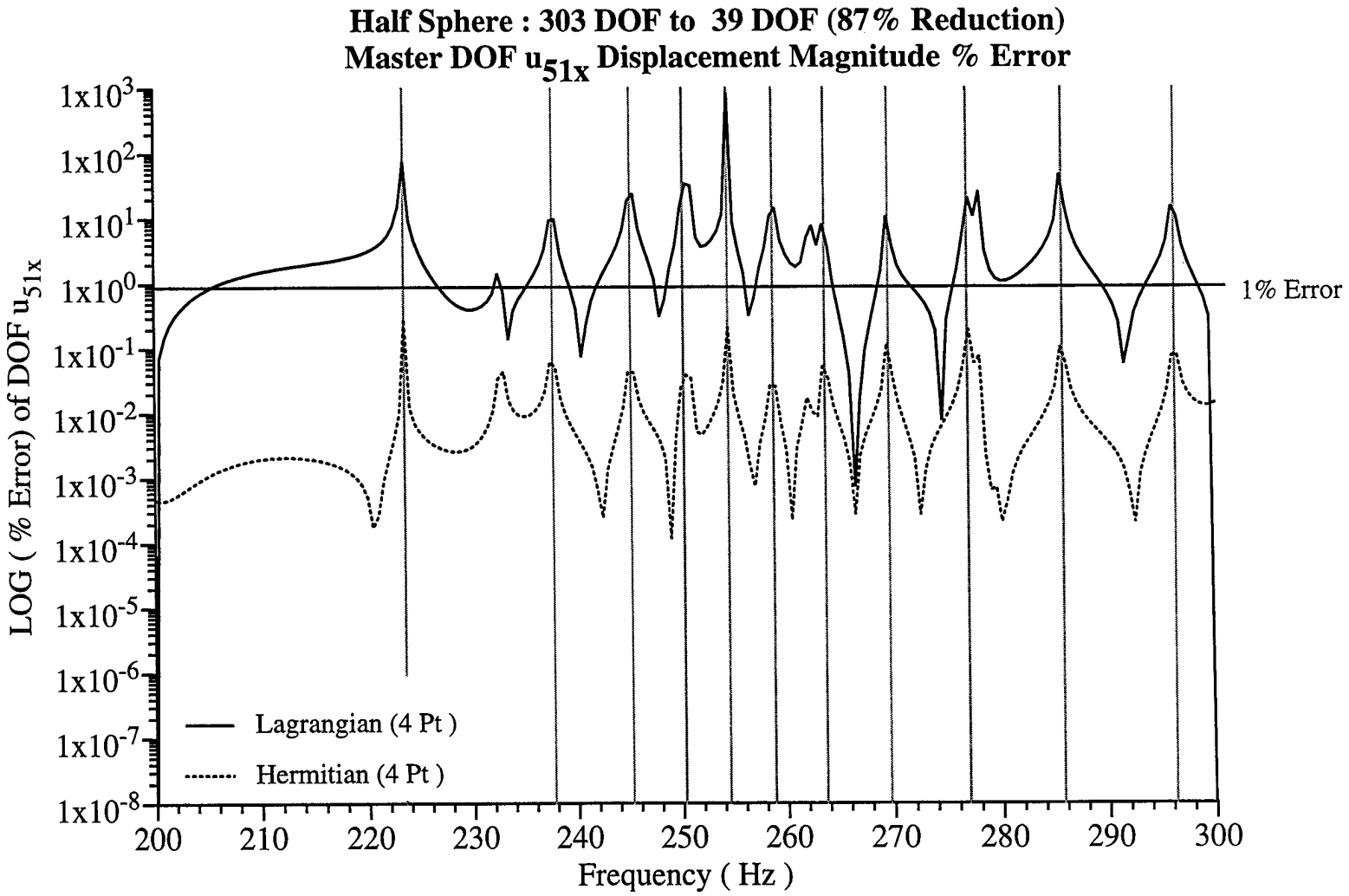


Figure 8.3.24 – % Error for the "master" DOF $|u|_{51x}$ for the frequency range of 200 to 300 Hz. NOTE: 87% DOF Reduction with less than 1% Error.

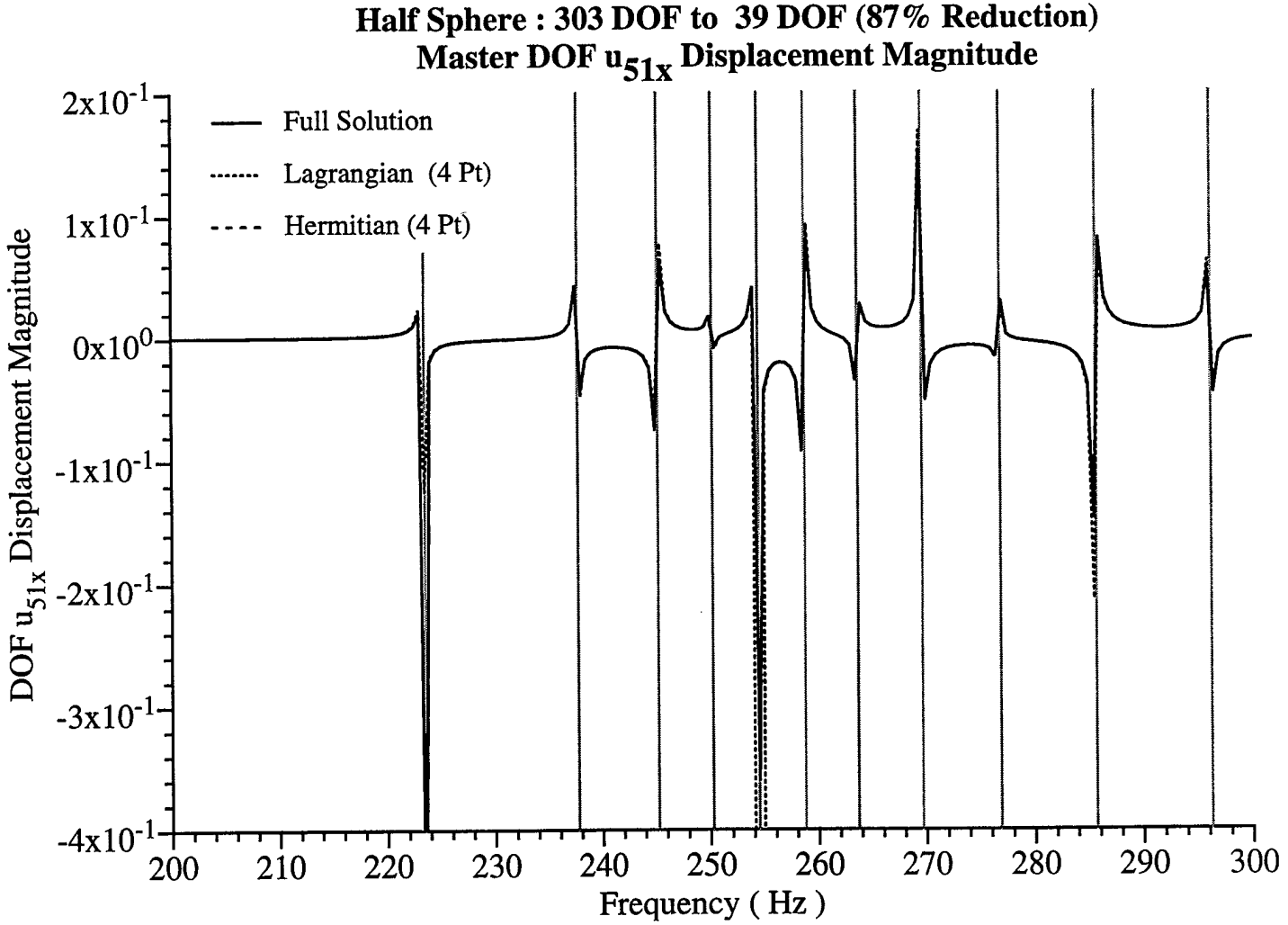


Figure 8.3.25 The "master" DOF $|u|_{51x}$ (magnitude of the displacement). NOTE: 87% DOF Reduction.

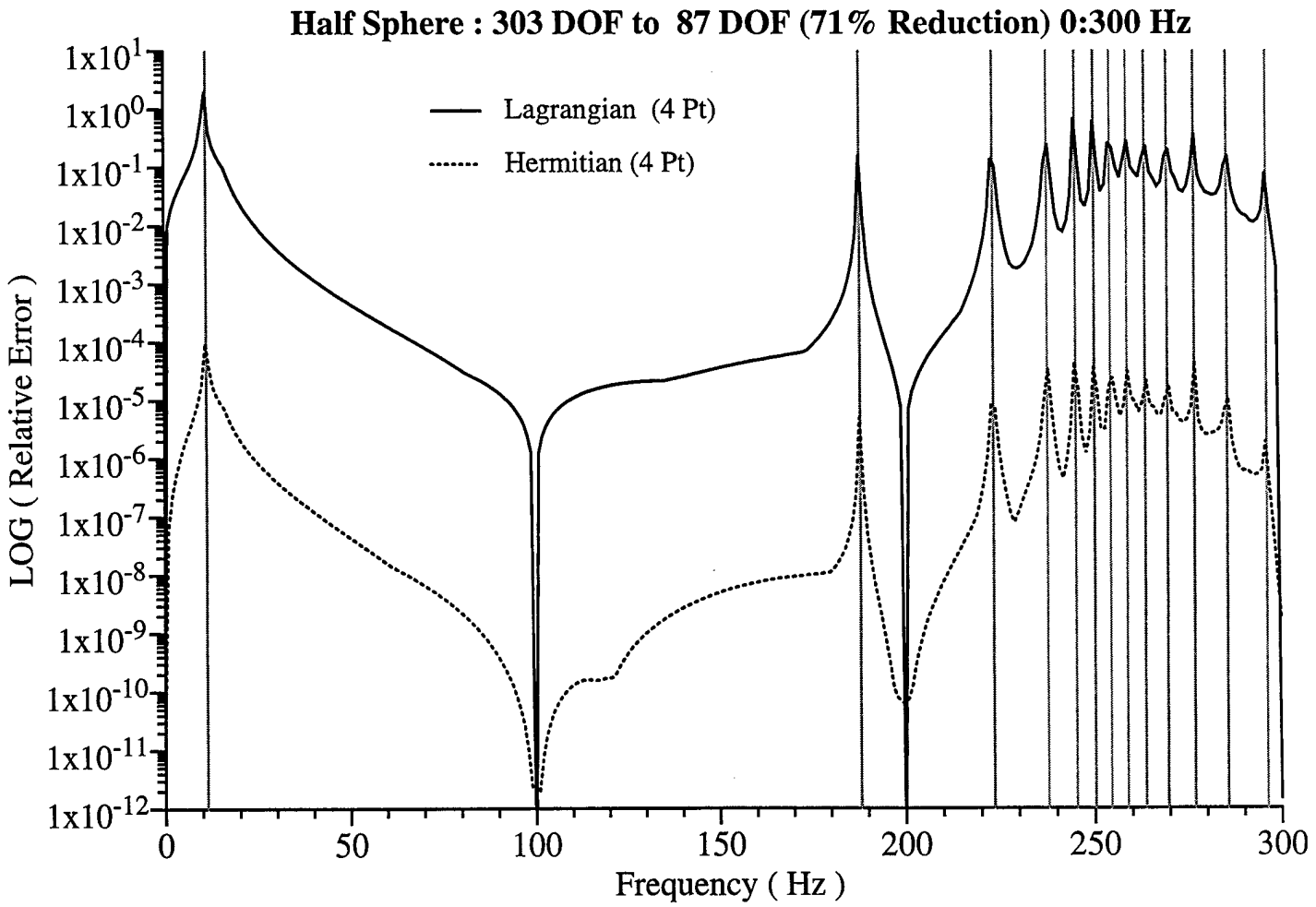


Figure 8.3.26 – Relative Error comparing different interpolation functions for the frequency range of 0 to 300 Hz and a frequency step size of 1 Hz.

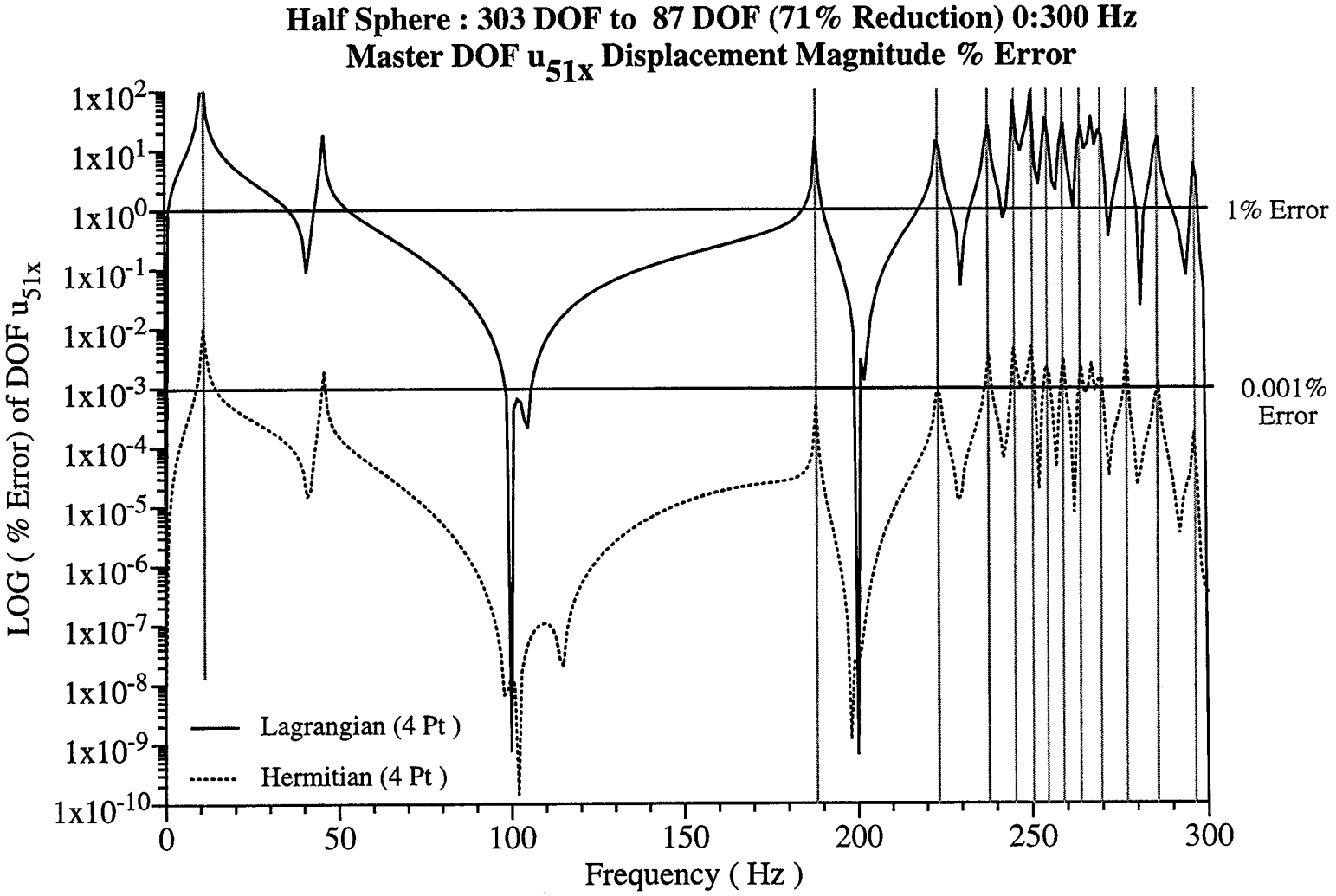


Figure 8.3.27 – % Error for the "master" DOF $|u|_{51x}$ for the frequency range of 0 to 300 Hz.

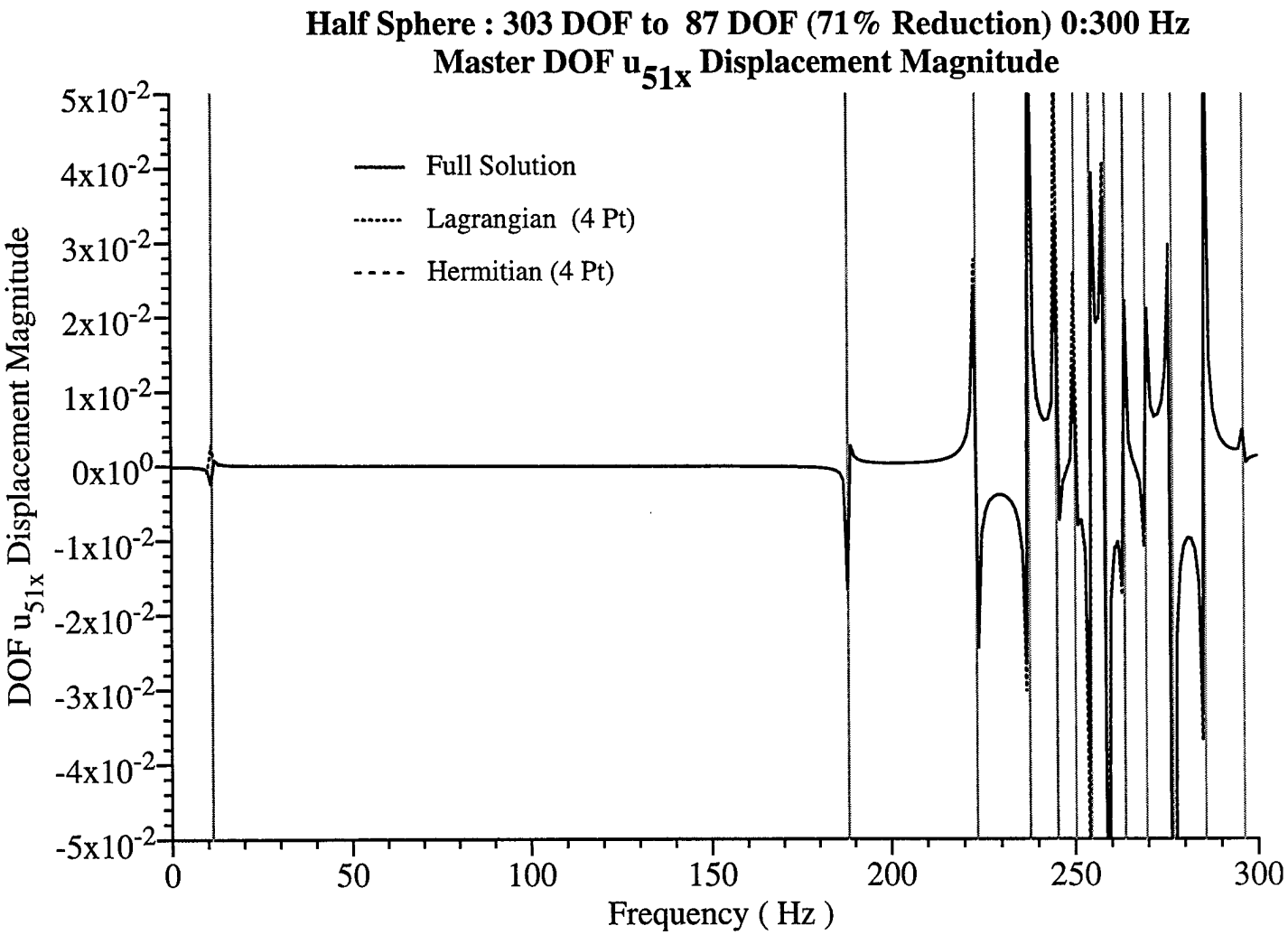


Figure 8.3.28 The "master" DOF $|u|_{51x}$ for the frequency range of 0 to 300 Hz.

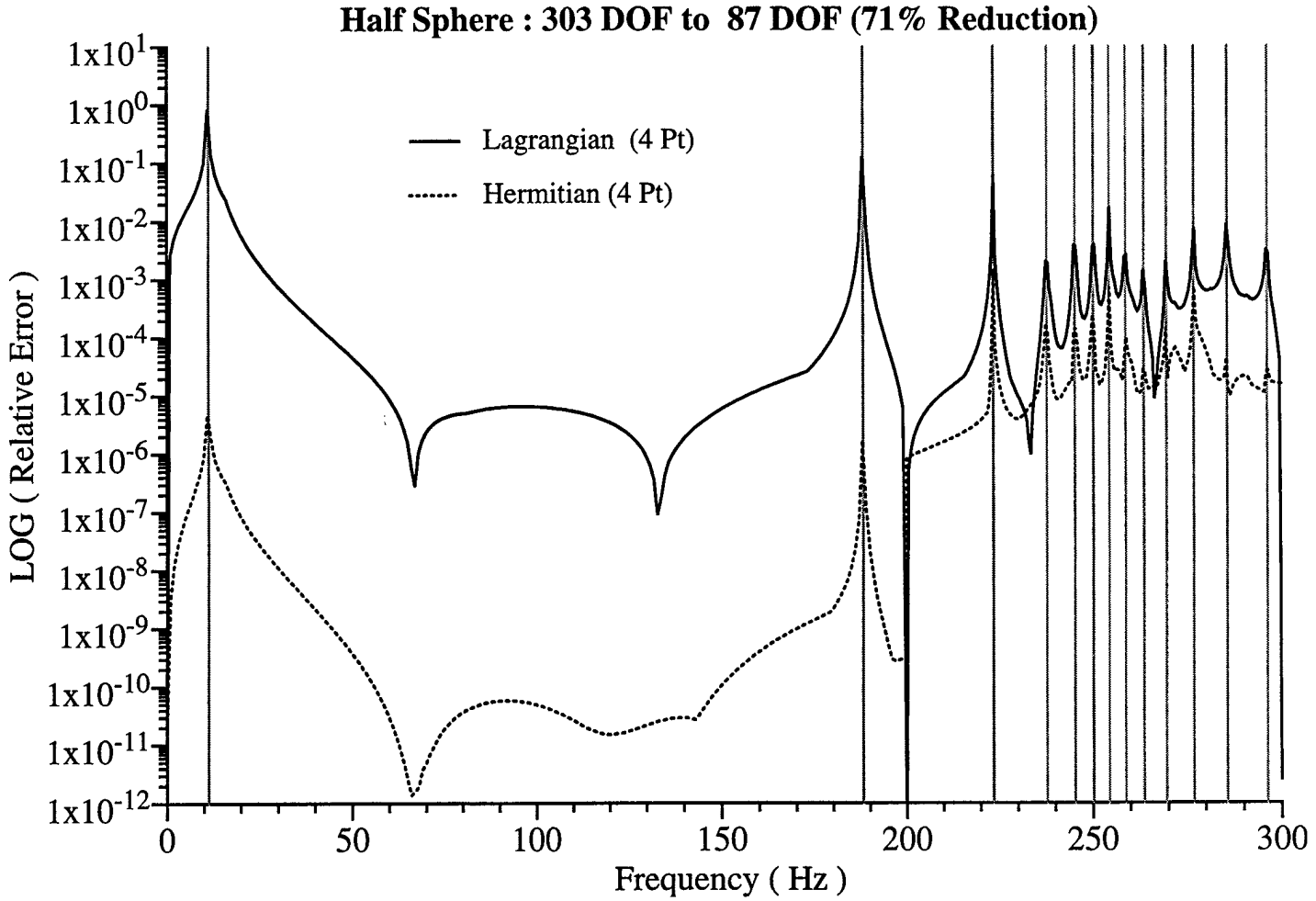


Figure 8.3.29 – Relative Error of combined data 0 to 200 Hz and 200 to 300 Hz.

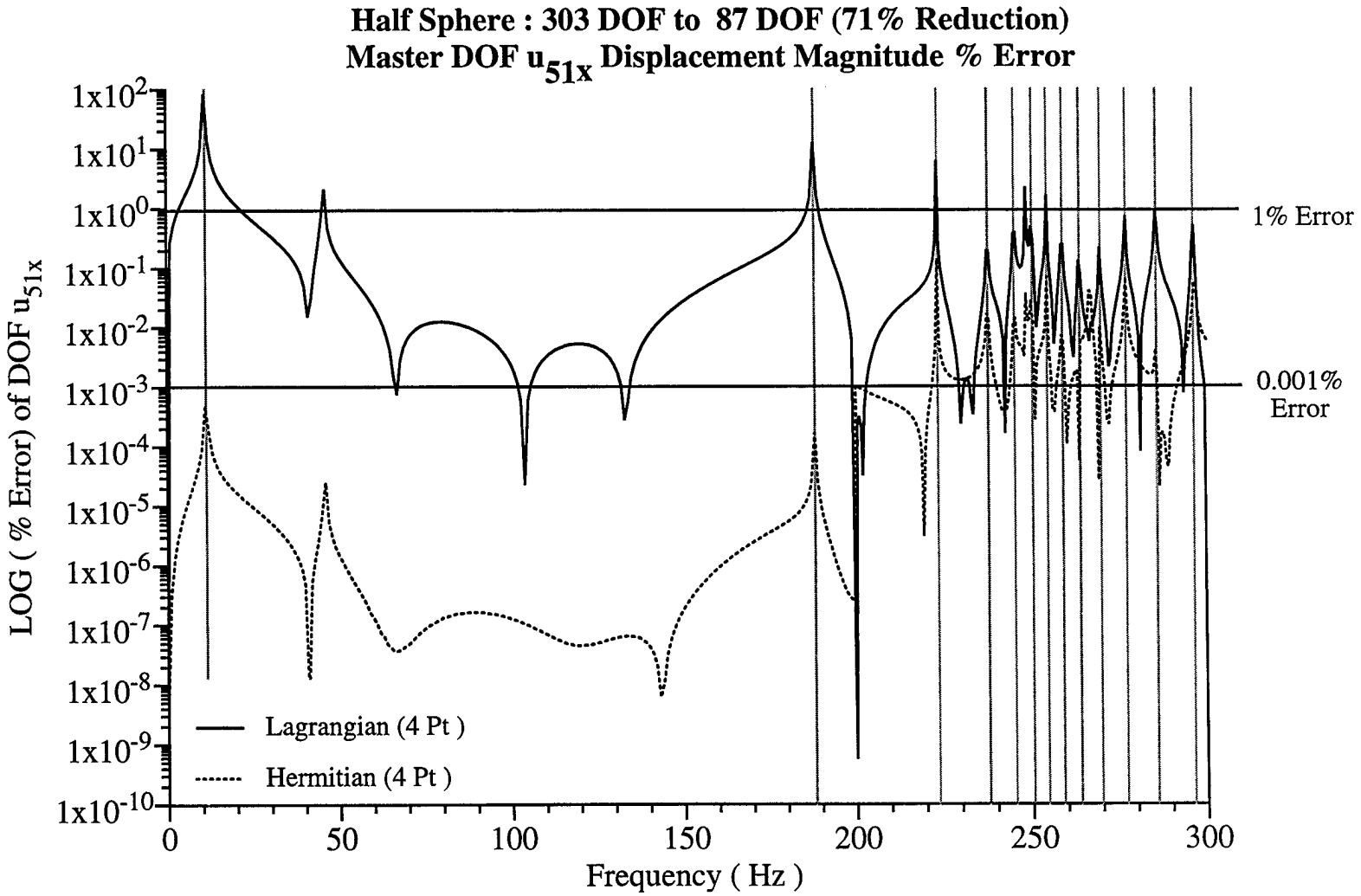


Figure 8.3.30 – % Error of combined data 0 to 200 Hz and 200 to 300 Hz at 71% DOF reduction.

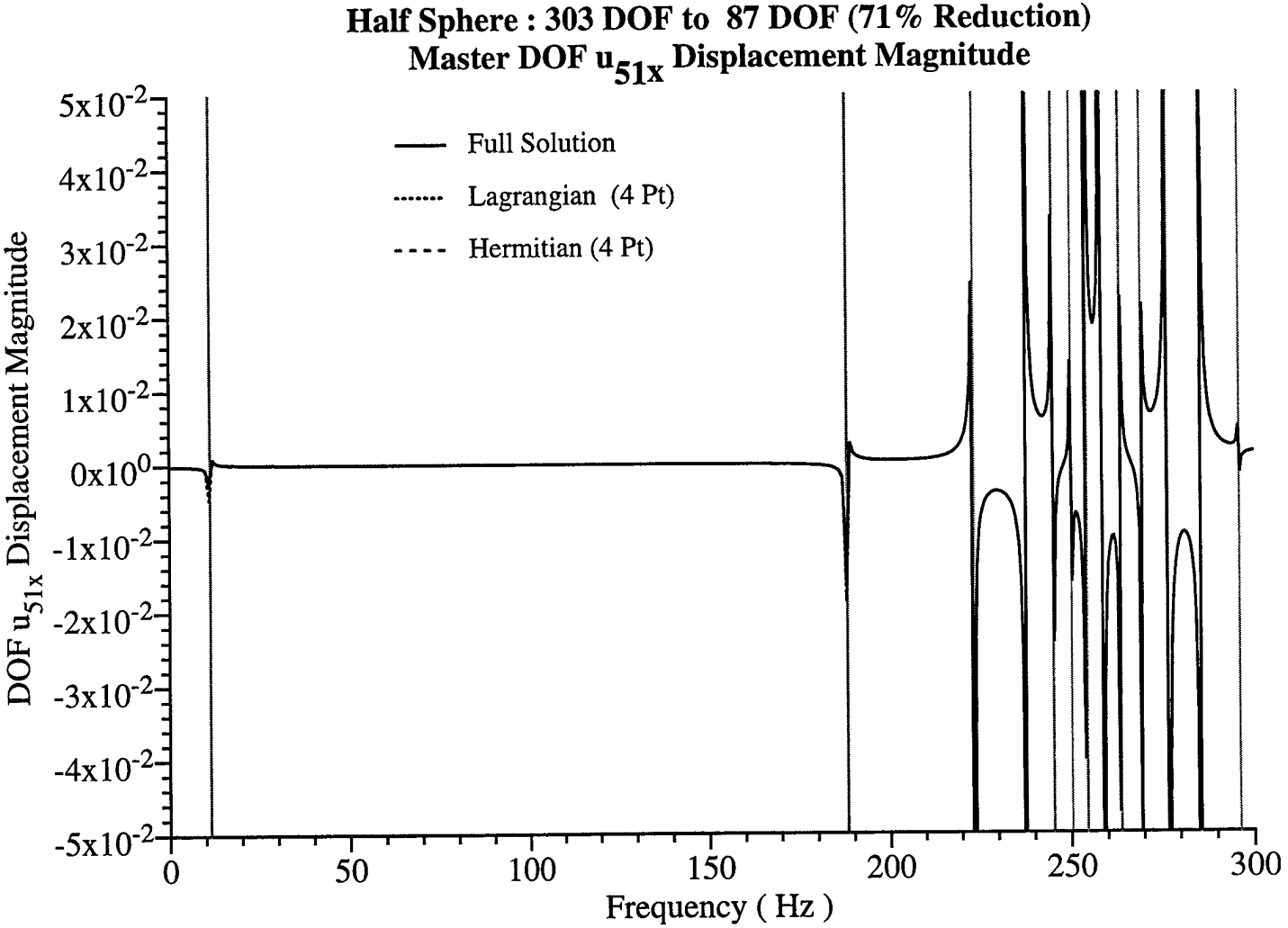


Figure 8.3.31 - The "master" DOF $|u|_{51x}$ for combined data 0 to 200 Hz and 200 to 300 Hz.

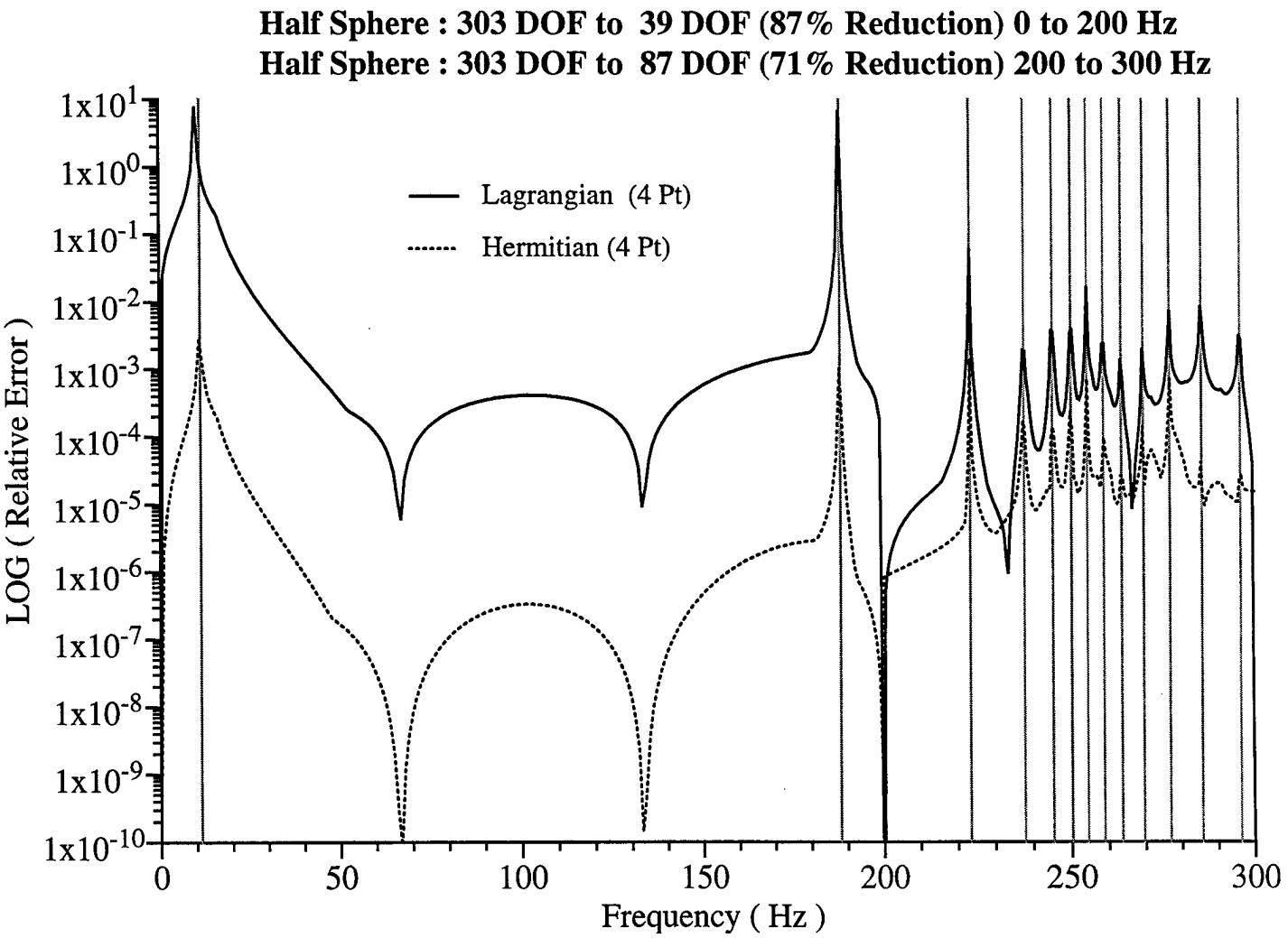


Figure 8.3.32 – Relative Error of combined data 0 to 200 Hz at 87% DOF reduction and 200 to 300 Hz at 71% DOF reduction.

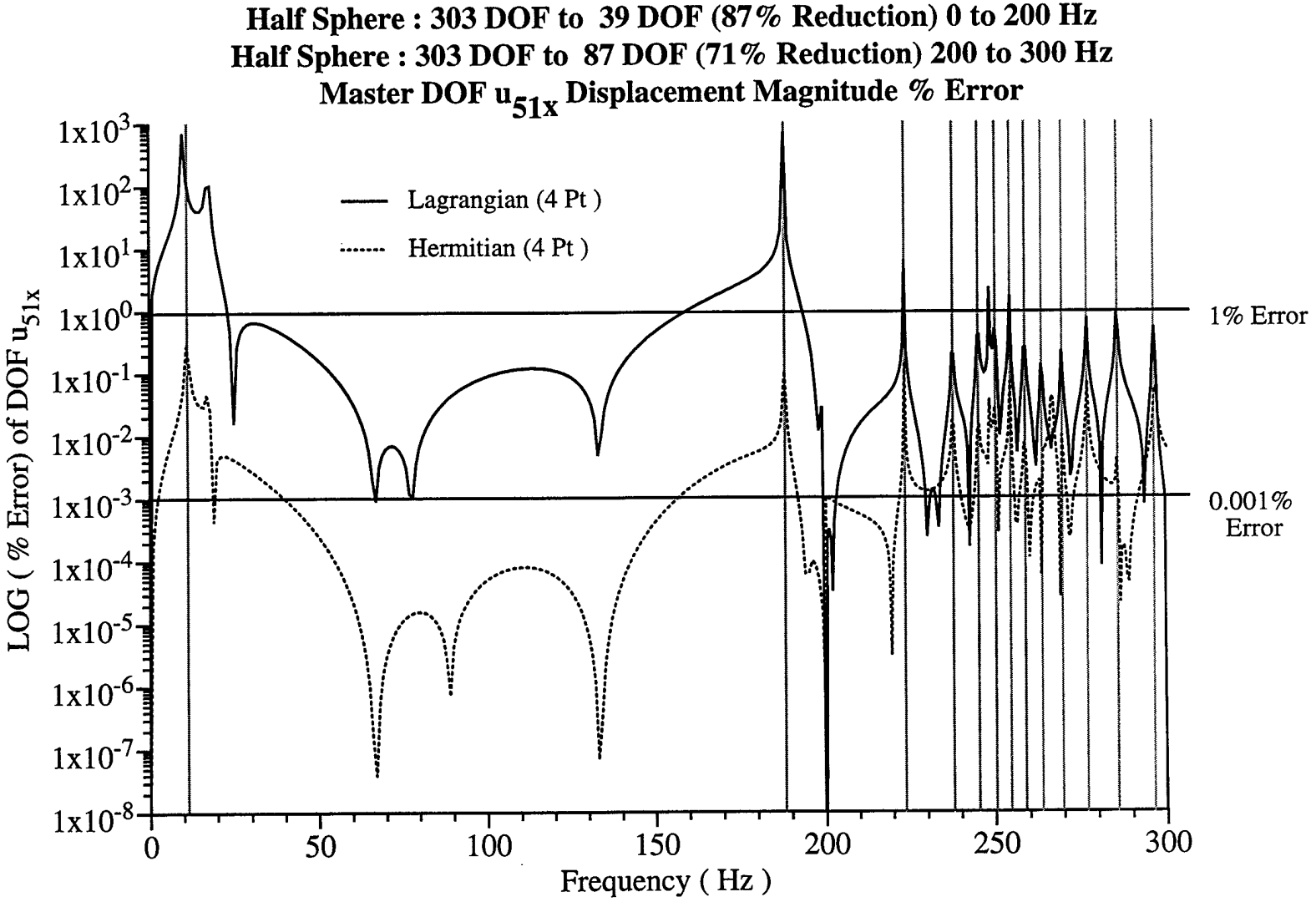


Figure 8.3.33 – % Error of combined data 0 to 200 Hz at 87% reduction and 200 to 300 Hz at 71% reduction.

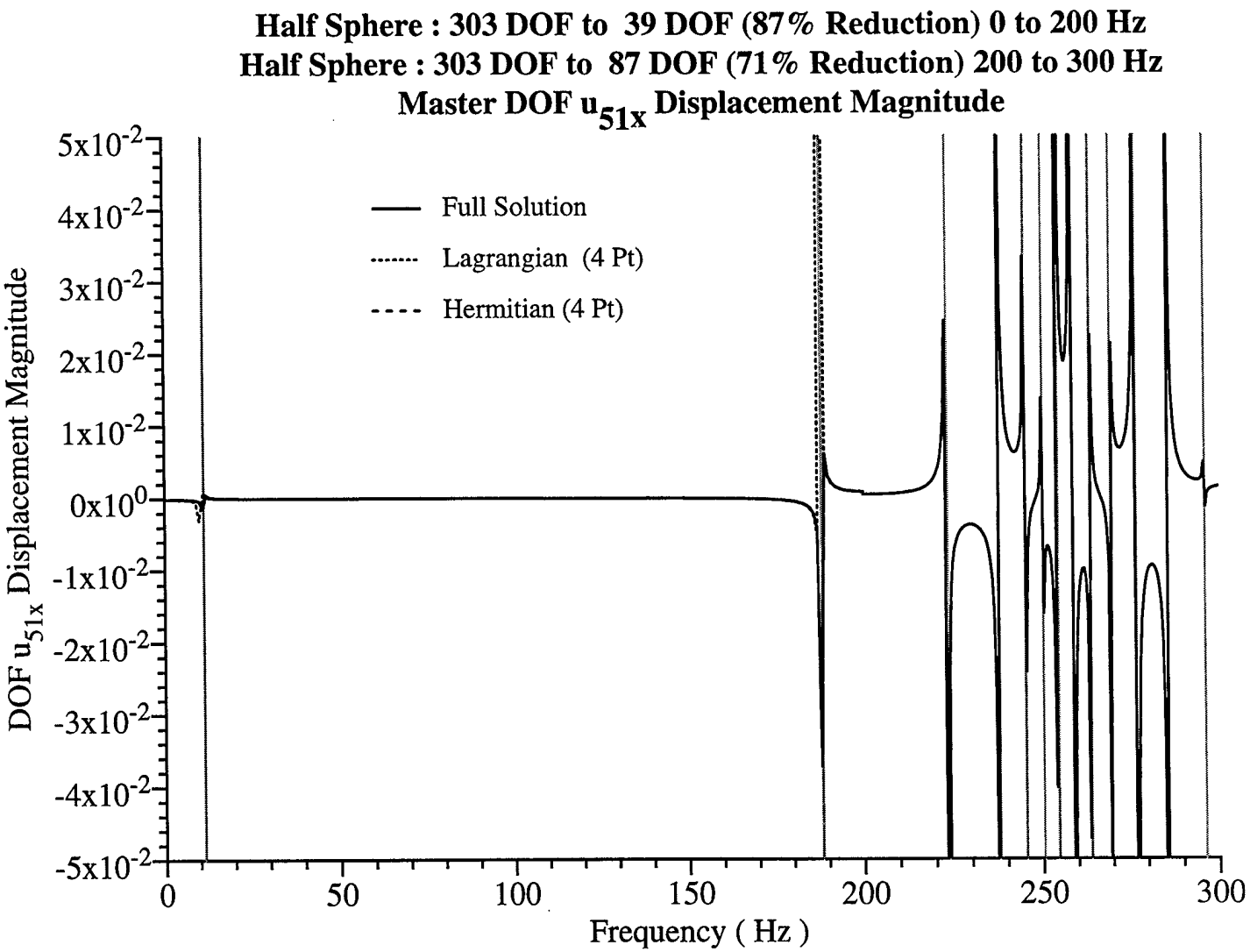


Figure 8.3.34 – The "master" DOF $|u|_{51x}$ for the combined data 0 to 200 Hz at 87% DOF reduction and 200 to 300 Hz at 71% DOF reduction.

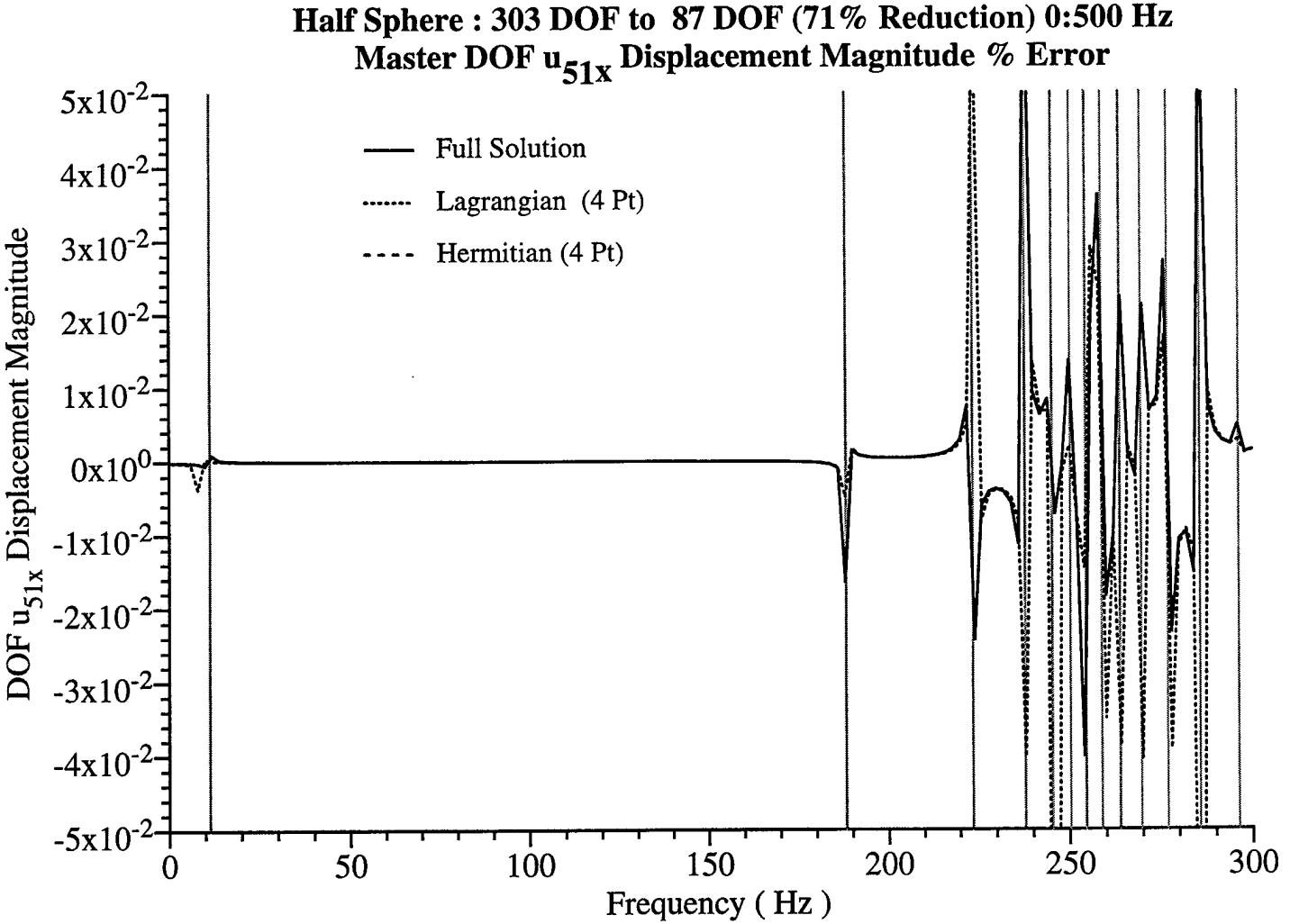


Figure 8.3.35 – The "master" DOF $|u|_{51x}$ for the data 0 to 500 Hz at 71% DOF reduction. NOTE: Expanded frequency scale from Figure 8.3.4.

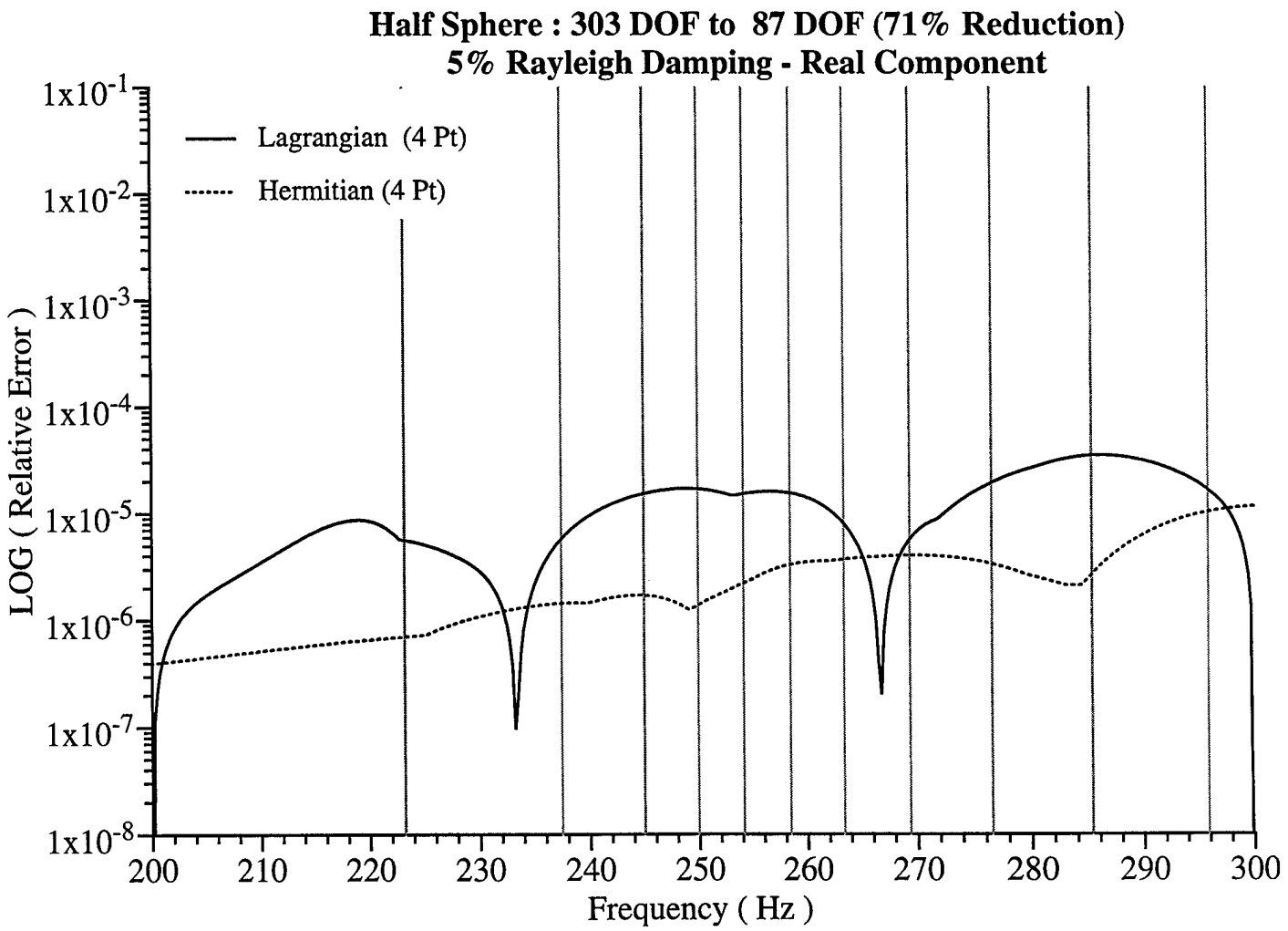


Figure 8.3.36 – Relative Error, real component, comparing interpolations for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz. NOTE: 5% Damping.

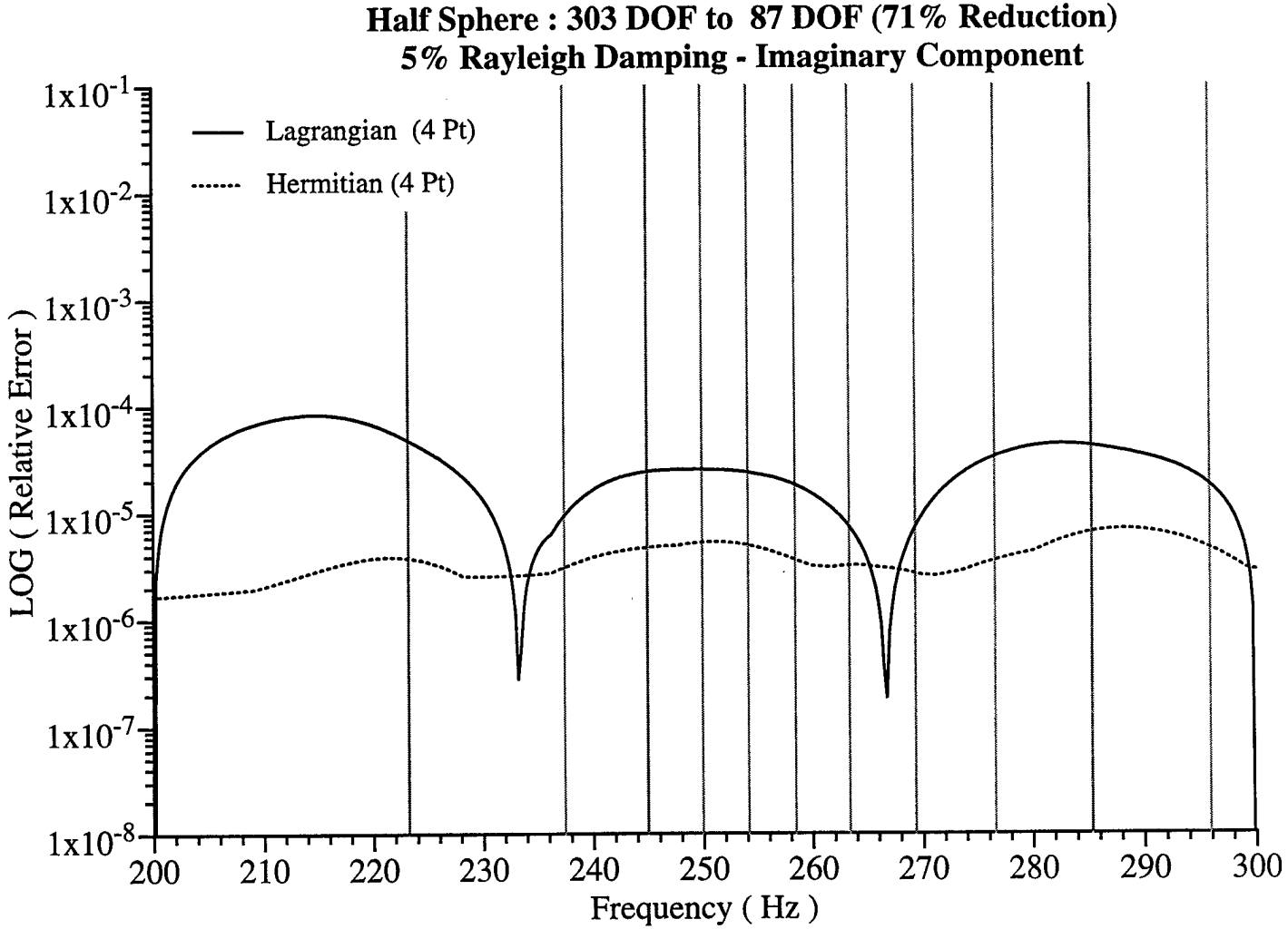


Figure 8.3.37 – Relative Error, imaginary component, comparing interpolations for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz. NOTE: 5% Damping.

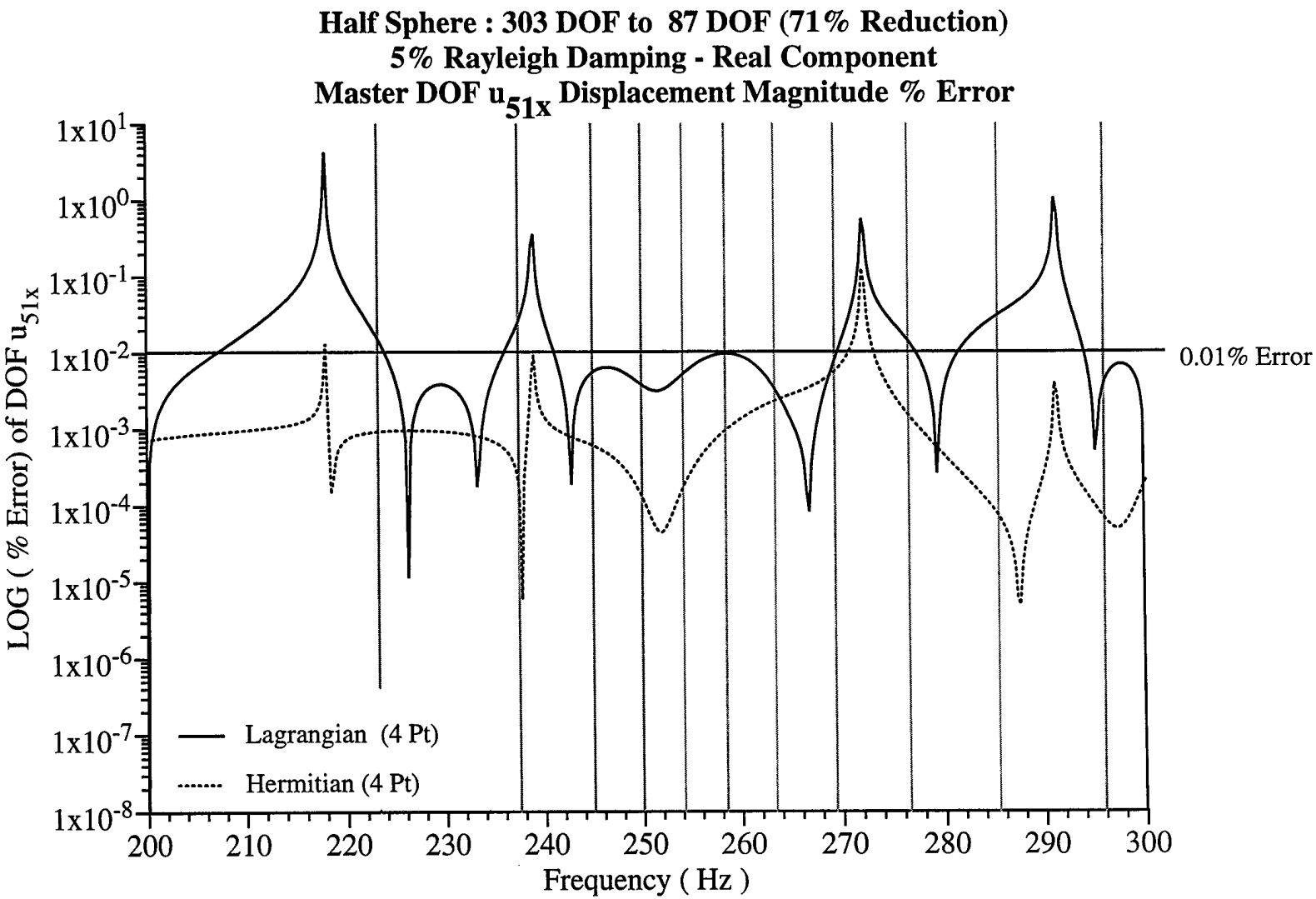


Figure 8.3.38 – % Error, real component, for the "master" DOF $|u|_{51x}$ comparing interpolations for the frequency range of 200 to 300 Hz. NOTE: 5% Damping with less than 0.01% Error.

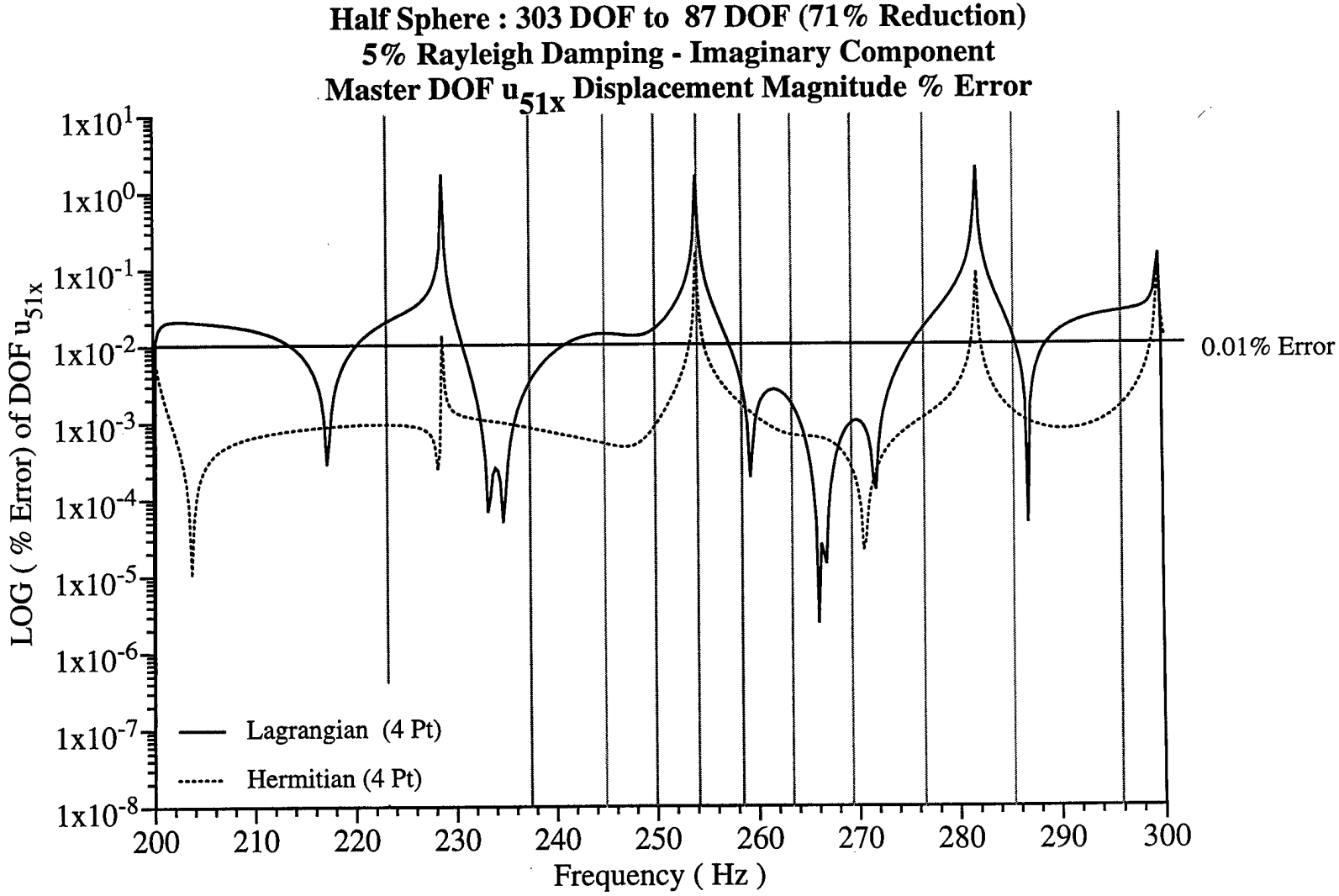


Figure 8.3.39 – % Error, imaginary component, for the "master" DOF $|u|_{51x}$ comparing interpolations for the frequency range of 200 to 300 Hz. NOTE: 5% of critical damping.

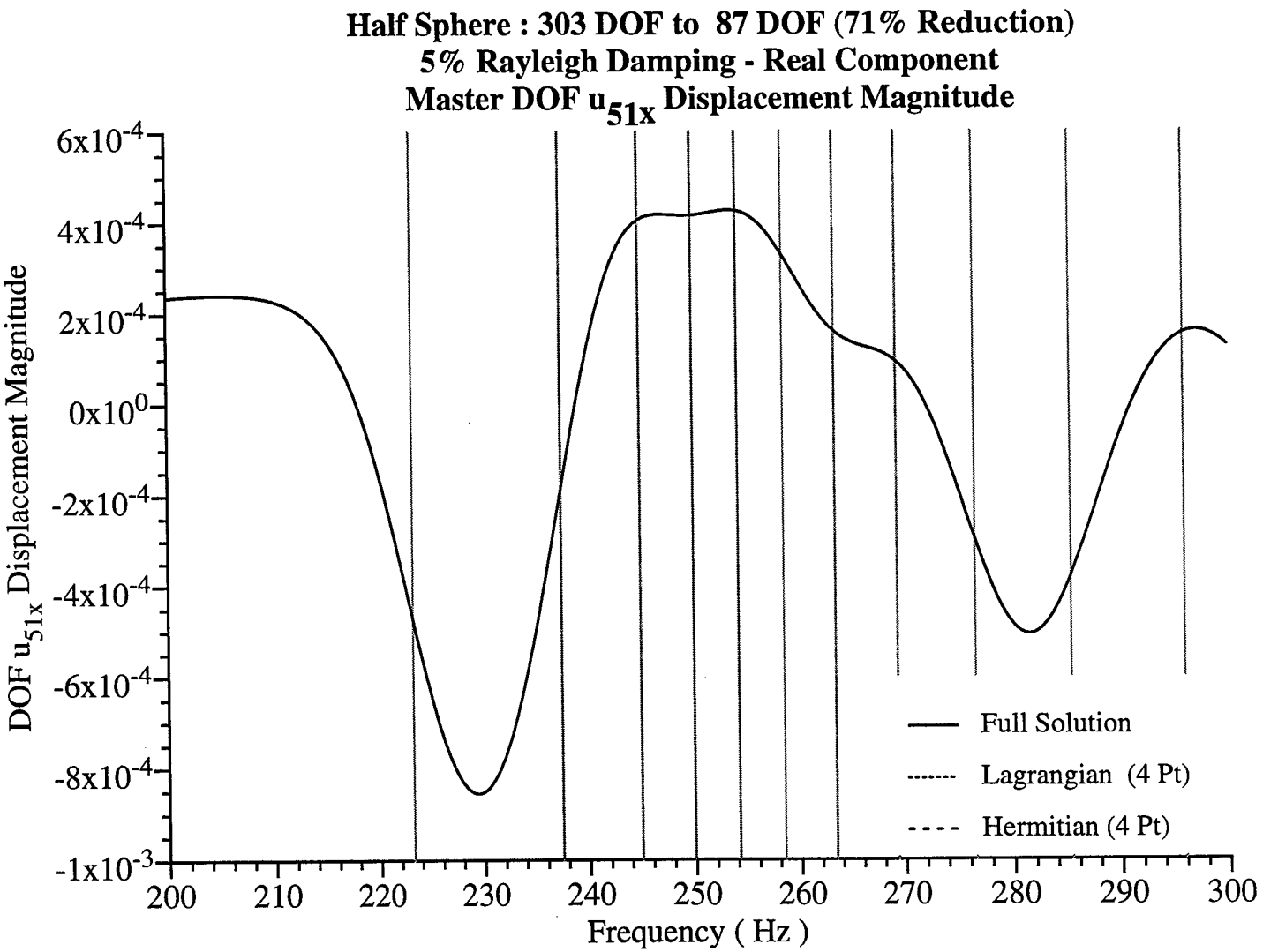


Figure 8.3.40 – The real component of the "master" DOF $|u|_{51x}$ for the range of 200 to 300 Hz and a step size of 0.5 Hz. NOTE: 5% Damping.

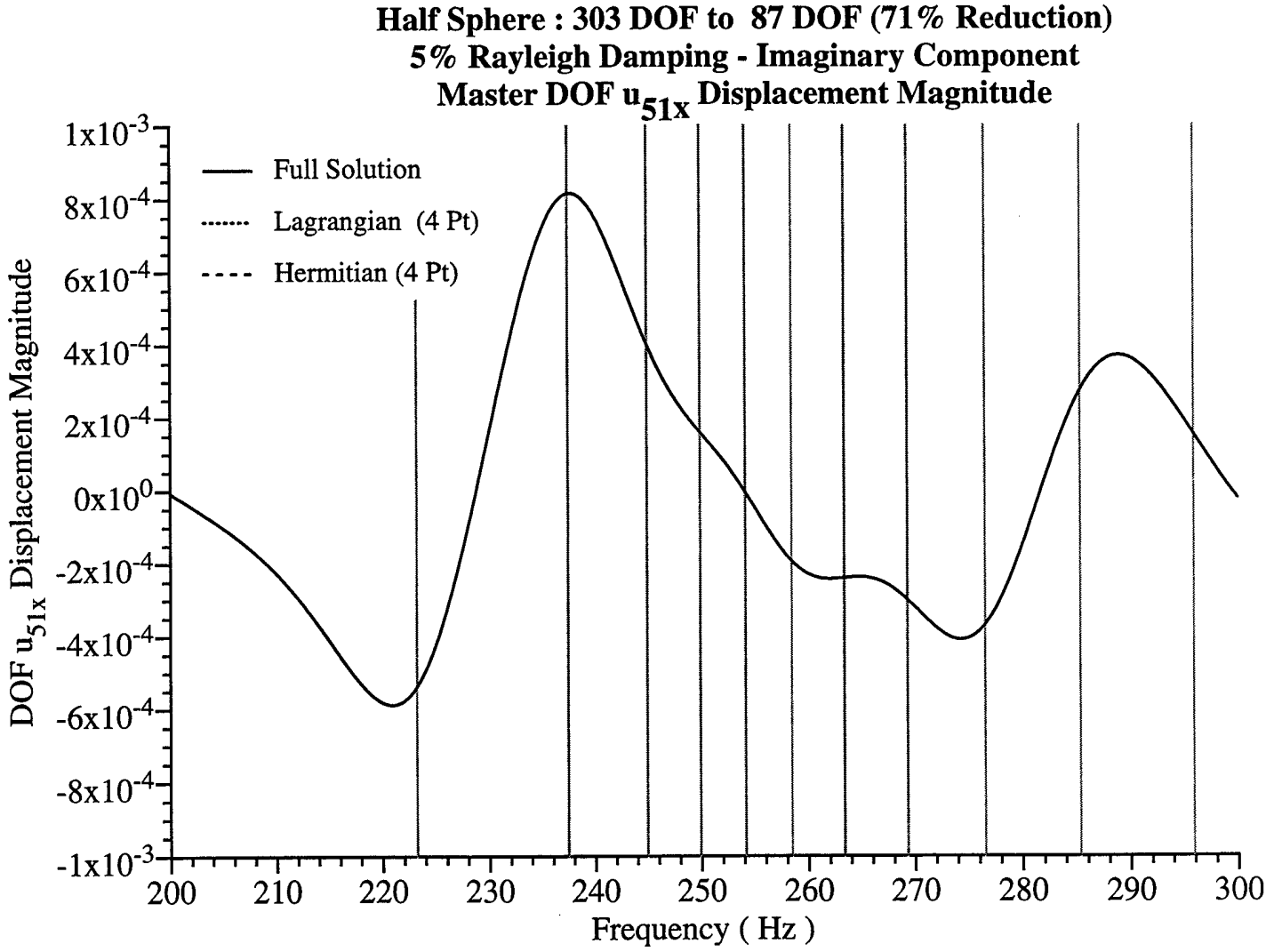


Figure 8.3.41 – The imaginary component of the "master" DOF $|u|_{51x}$ for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz. NOTE: 5% Damping.

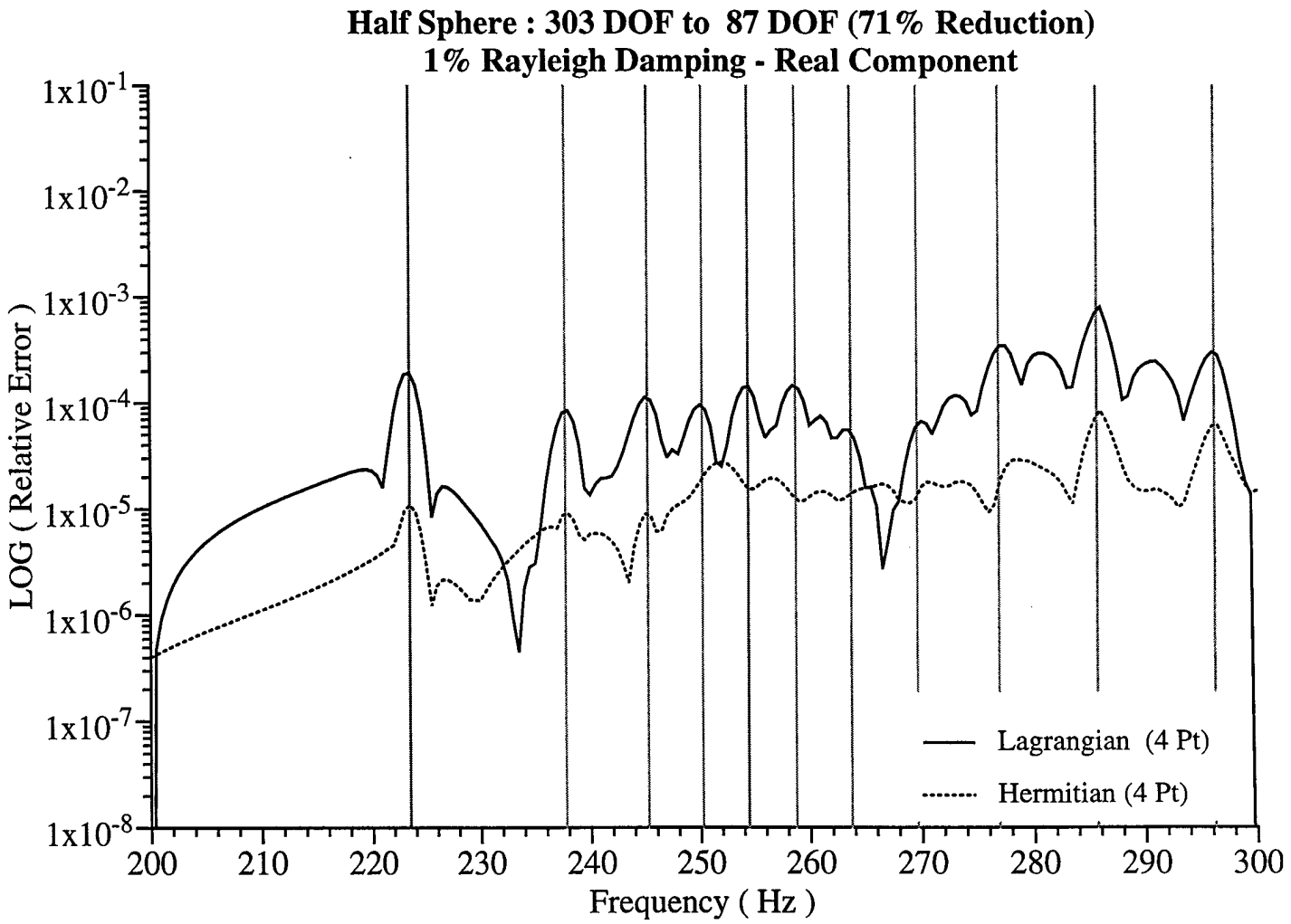


Figure 8.3.42 – Relative Error, real component, comparing interpolations for the frequency range of 200 to 300 Hz and a frequency step size of 0.25 Hz. NOTE: 1% Damping.

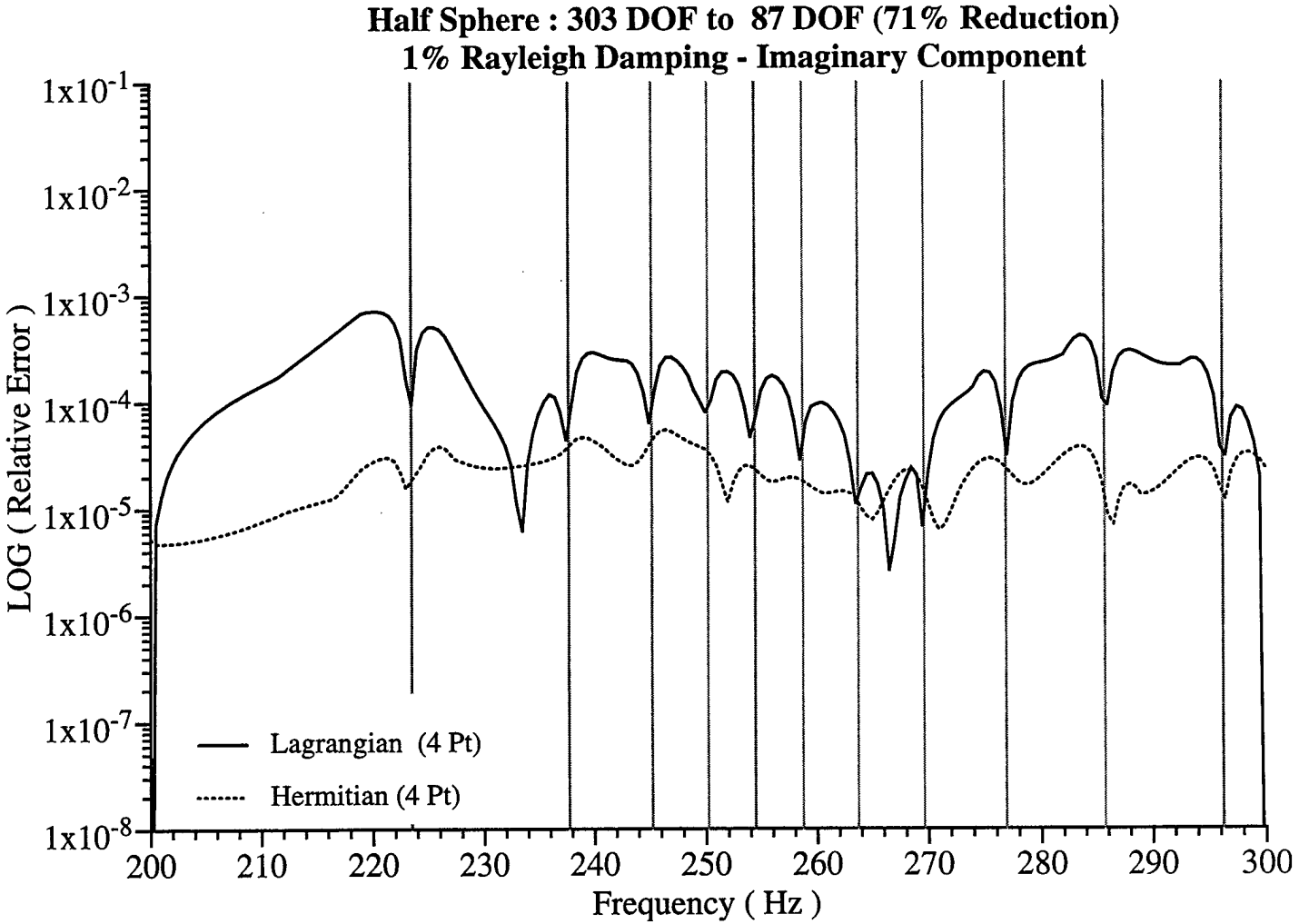


Figure 8.3.43 – Relative Error, imaginary component, comparing interpolations for the frequency range of 200 to 300 Hz and a frequency step size of 0.25 Hz. NOTE: 1% of critical damping.

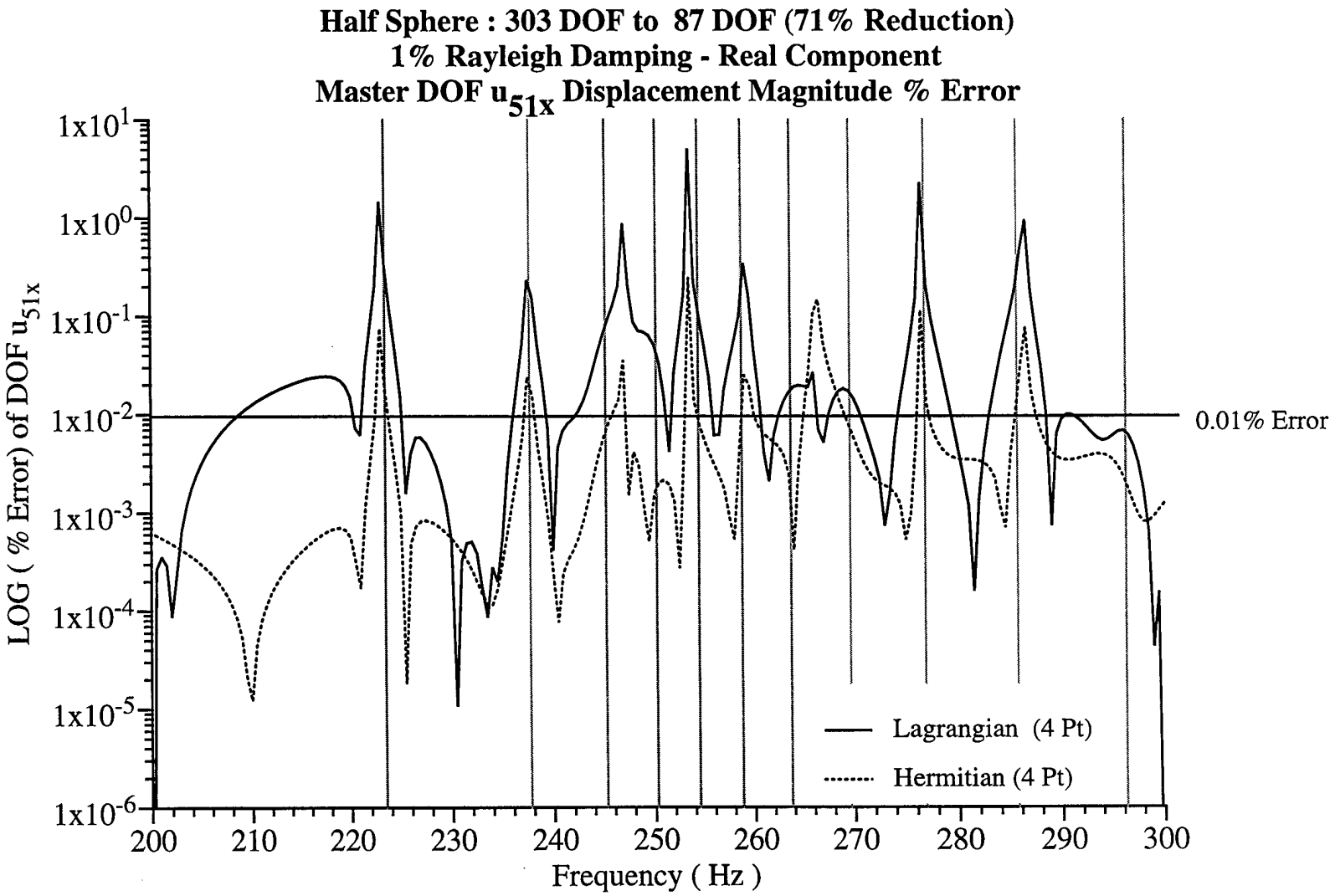


Figure 8.3.44 – % Error, real component, for the "master" DOF $|u|_{51x}$ comparing interpolations for the range of 200 to 300 Hz. NOTE: 1% Damping with less than 0.01% Error.

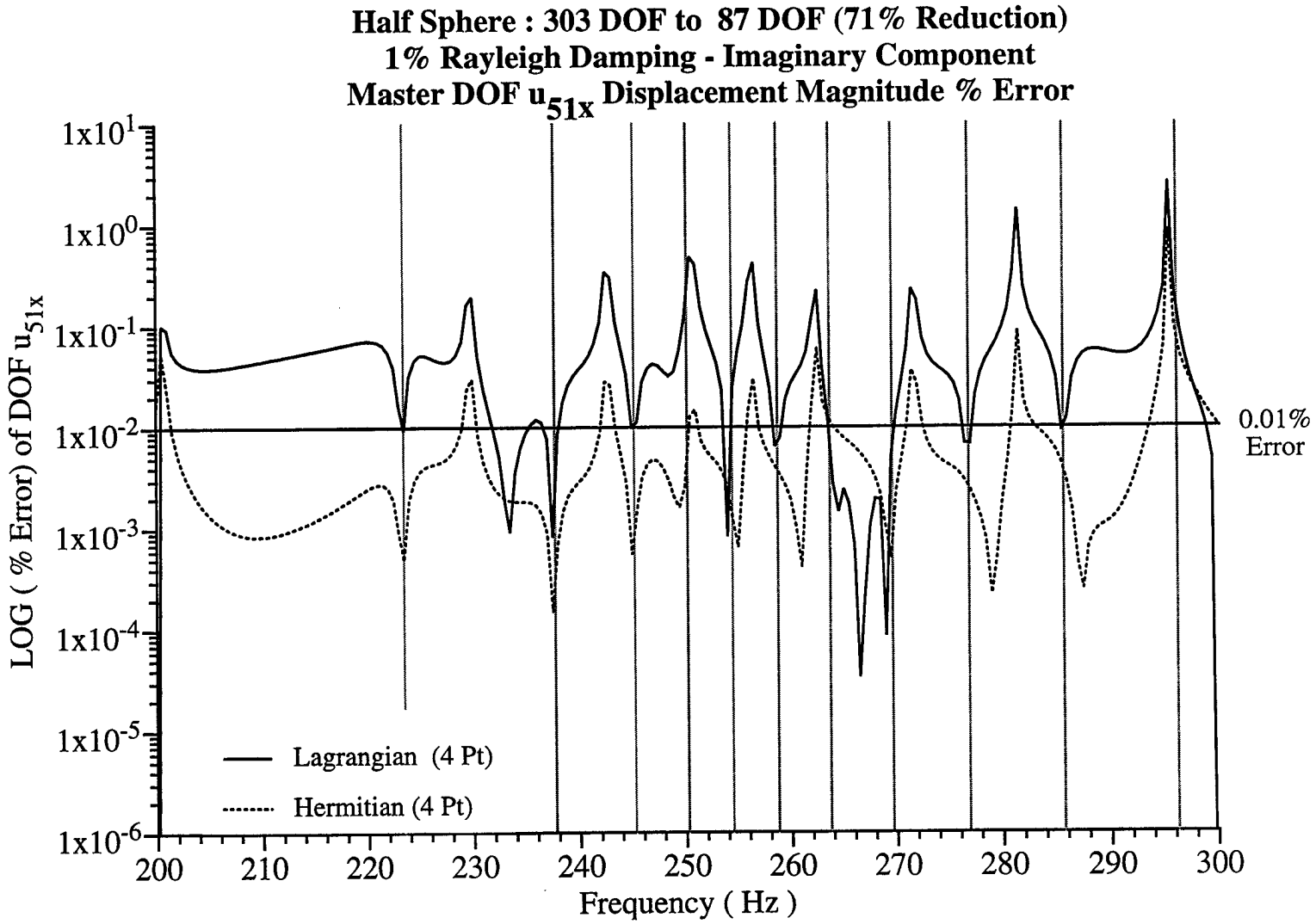


Figure 8.3.45 – % Error, imaginary component, for the "master" DOF lul_{51x} comparing interpolations for the frequency range of 200 to 300 Hz. NOTE: 1% Damping.

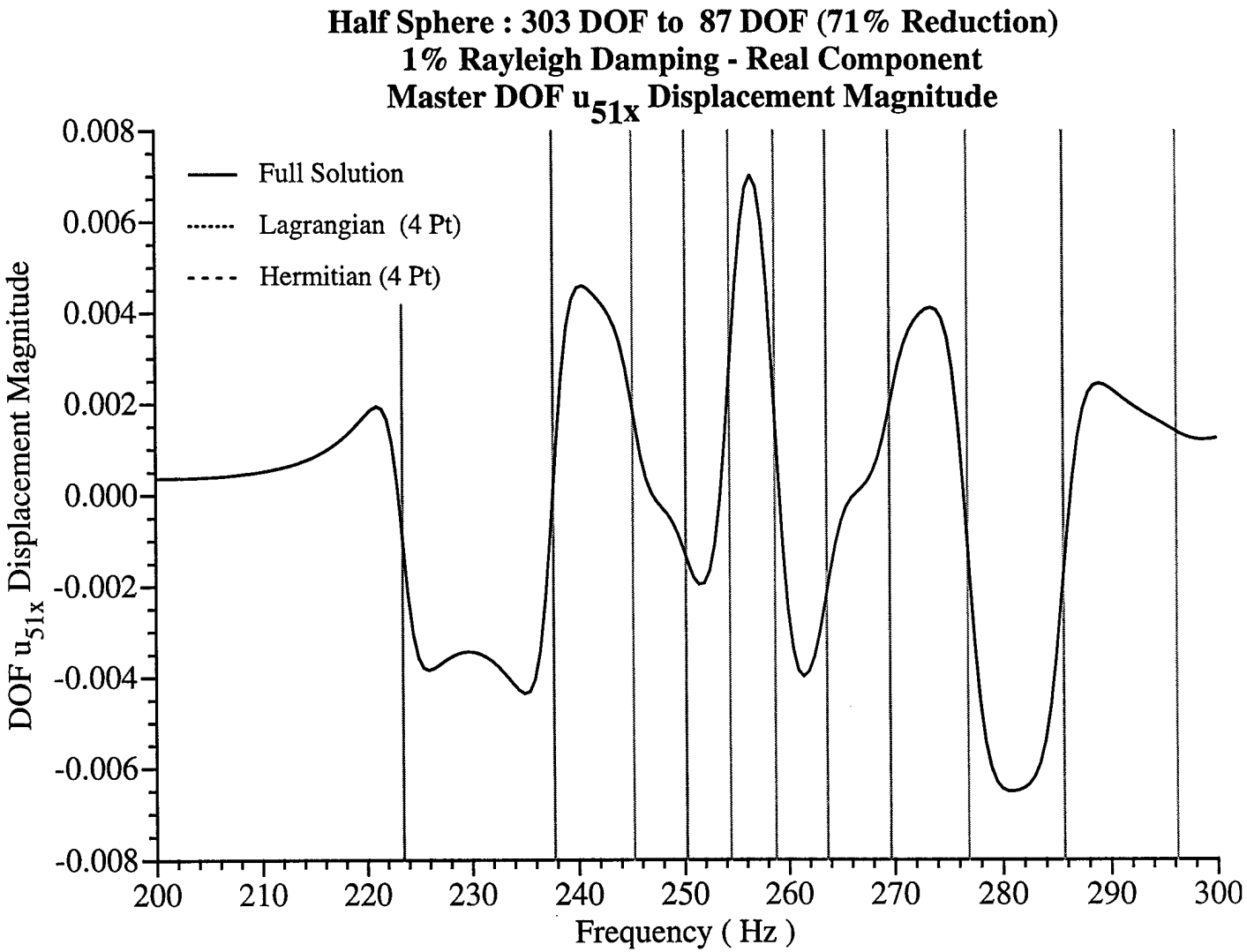


Figure 8.3.46 – The real component of the "master" DOF $|u|_{51x}$ for the frequency range of 200 to 300 Hz and a frequency step size of 0.25 Hz. NOTE: 1% Damping.

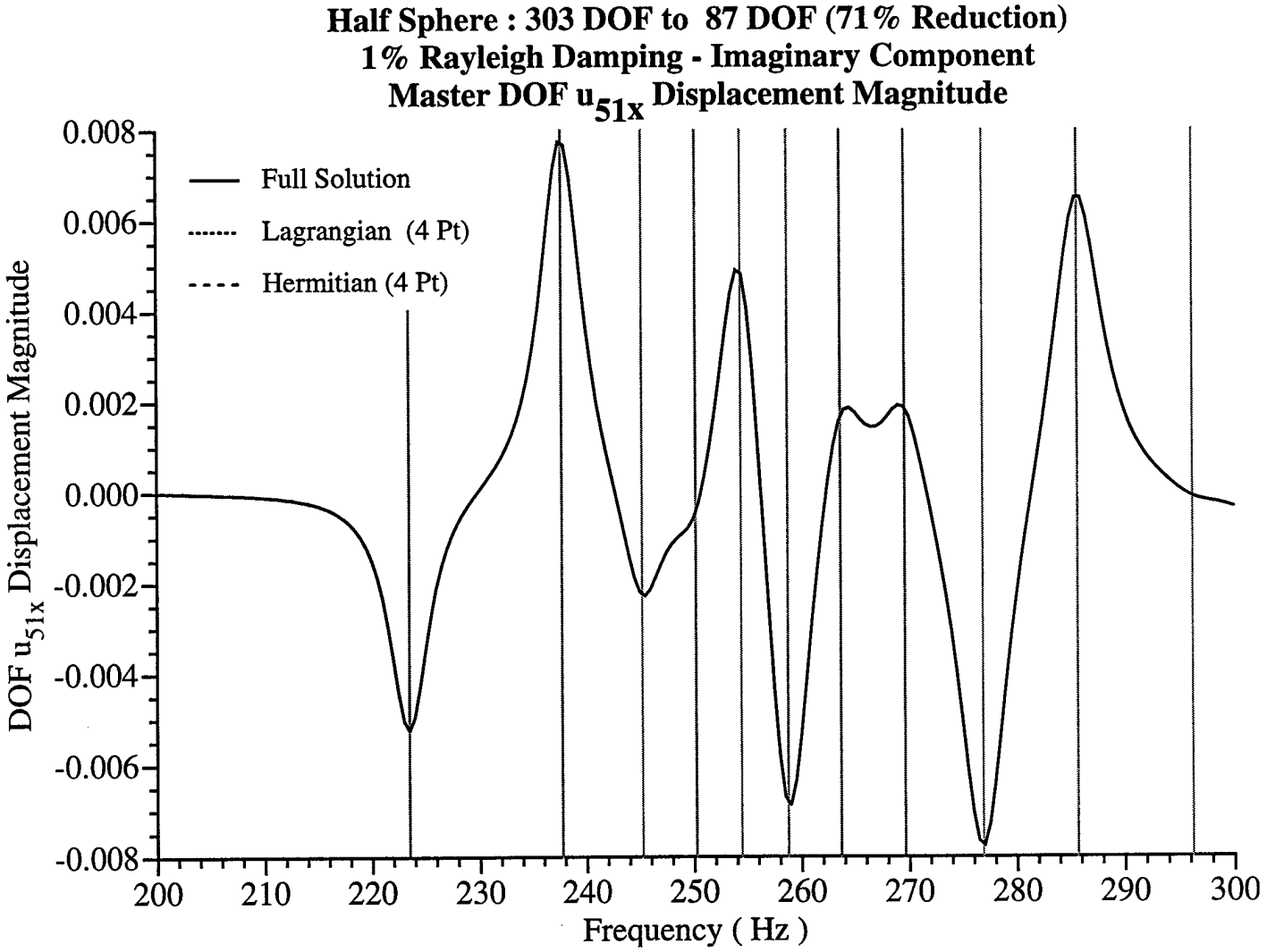


Figure 8.3.47 – The imaginary component of the "master" DOF Iul_{51x} for the frequency range of 200 to 300 Hz and a frequency step size of 0.25 Hz. NOTE: 1% Damping.

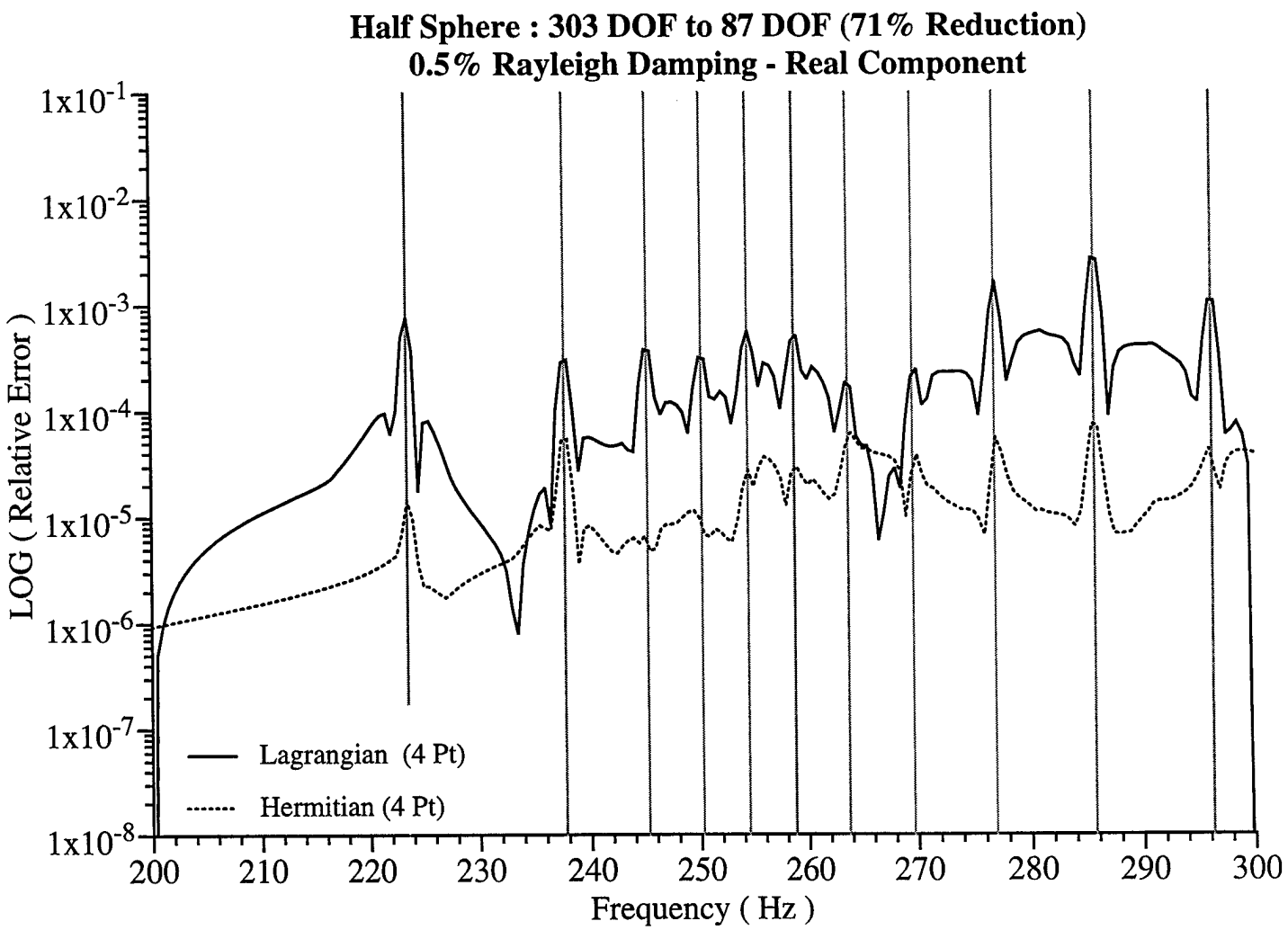


Figure 8.3.48 – Relative Error, real component, comparing interpolation functions for the frequency range of 200 to 300 Hz and a frequency step size of 0.25 Hz. NOTE: 0.5% of critical damping.

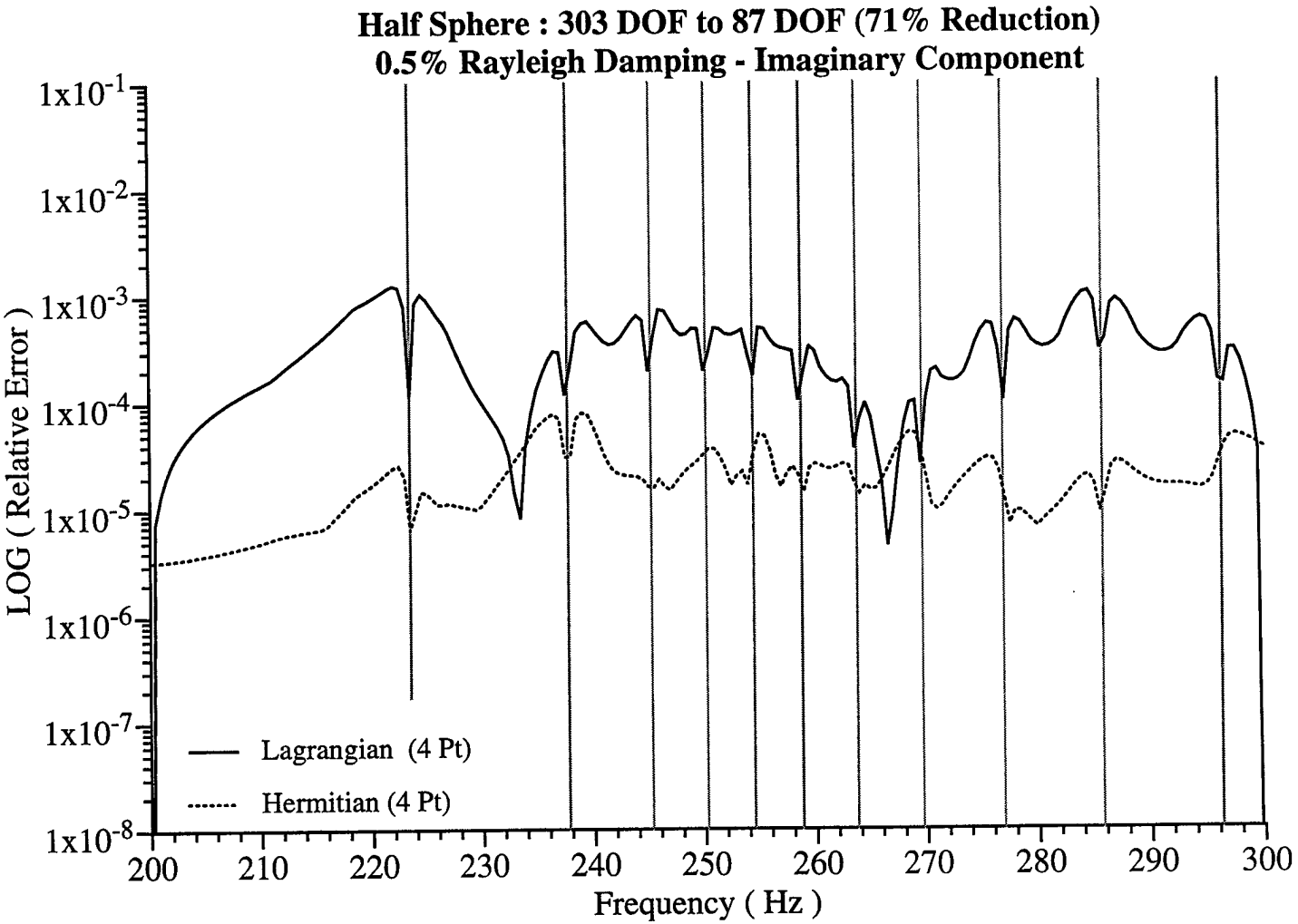


Figure 8.3.49 – Relative Error, imaginary component, comparing interpolation functions for the frequency range of 200 to 300 Hz and a frequency step size of 0.25 Hz. NOTE: 0.5% Damping.

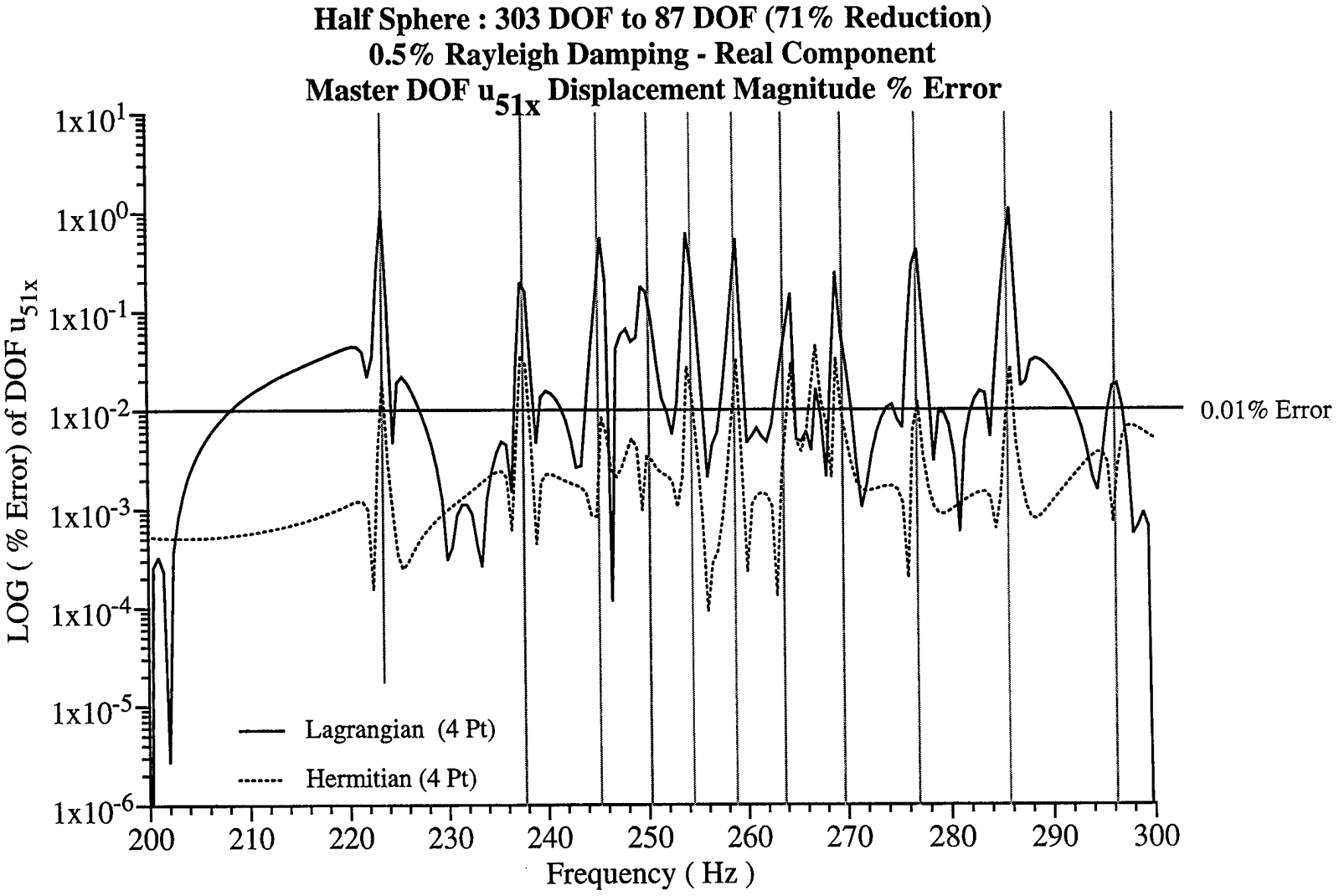


Figure 8.3.50 – % Error, real component, for the "master" DOF $|u|_{51x}$ comparing interpolations for the frequency range of 200 to 300 Hz. NOTE: 0.5% Damping with less than 0.01% Error.

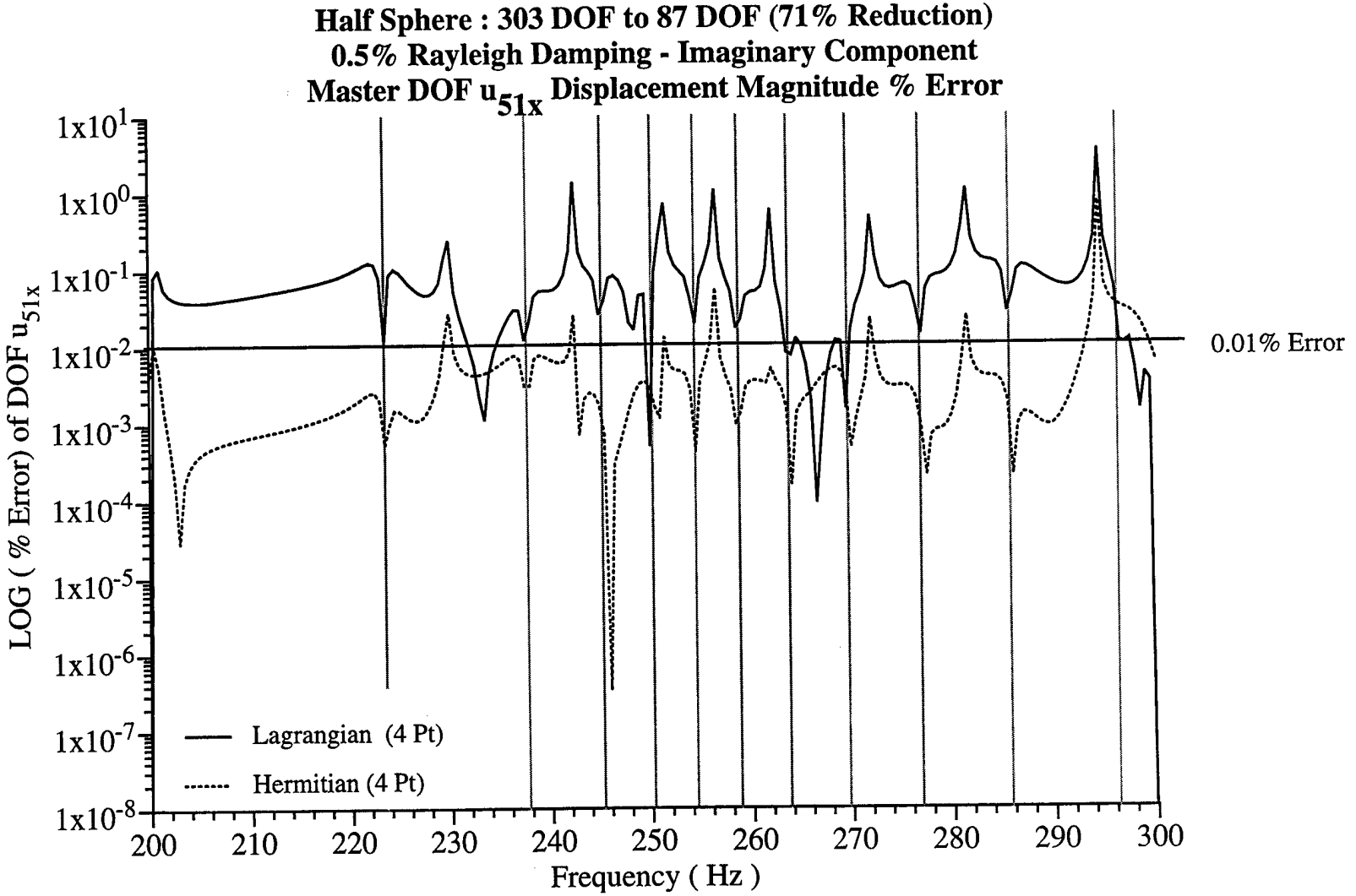


Figure 8.3.51 – % Error, imaginary component, for the "master" DOF $|u|_{51x}$ comparing interpolations for the frequency range of 200 to 300 Hz. NOTE: 0.5% Damping.

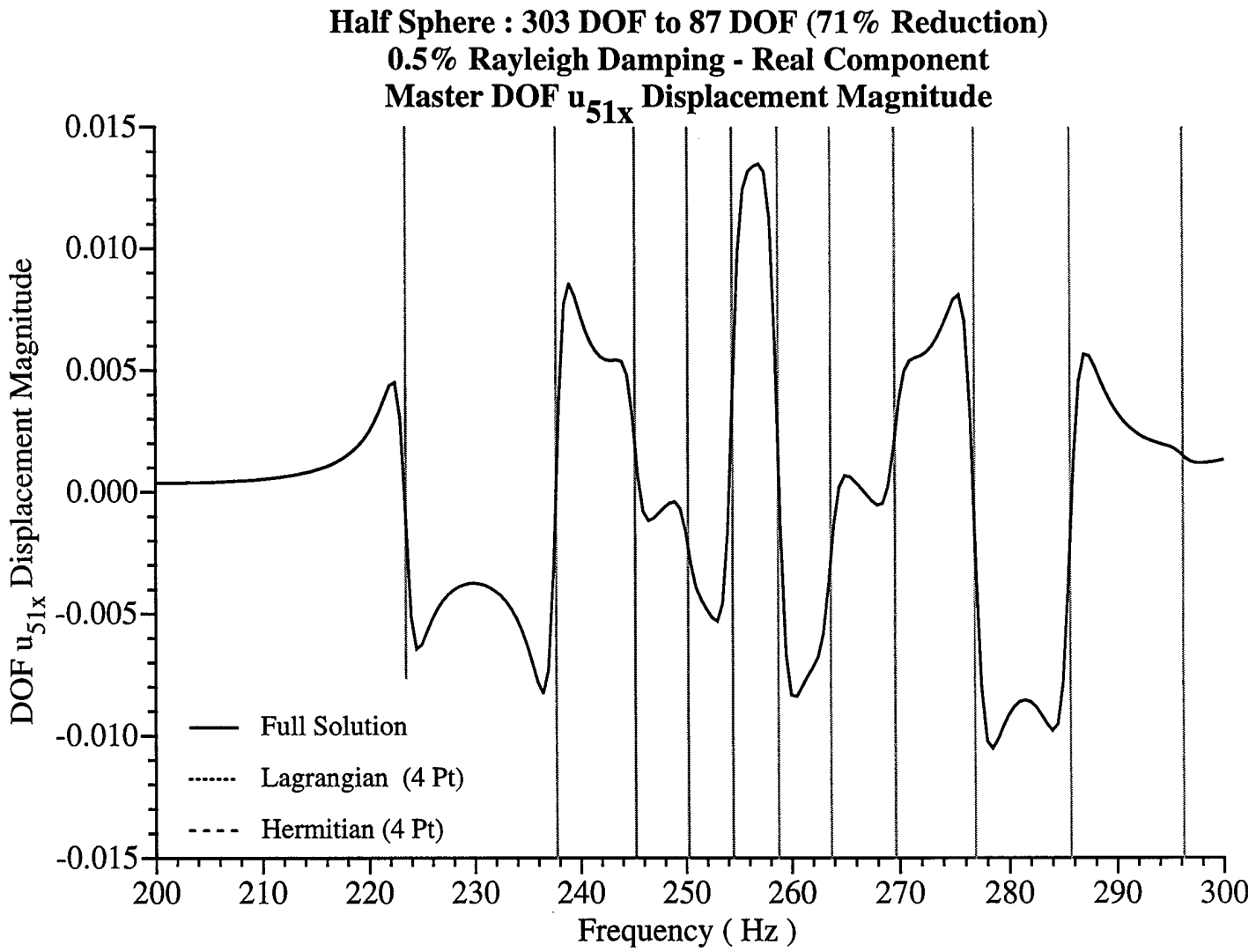


Figure 8.3.52 – The real component of the "master" DOF $|u_{51x}|$ for the frequency range of 200 to 300 Hz and a frequency step size of 0.25 Hz. NOTE: 0.5% Damping.

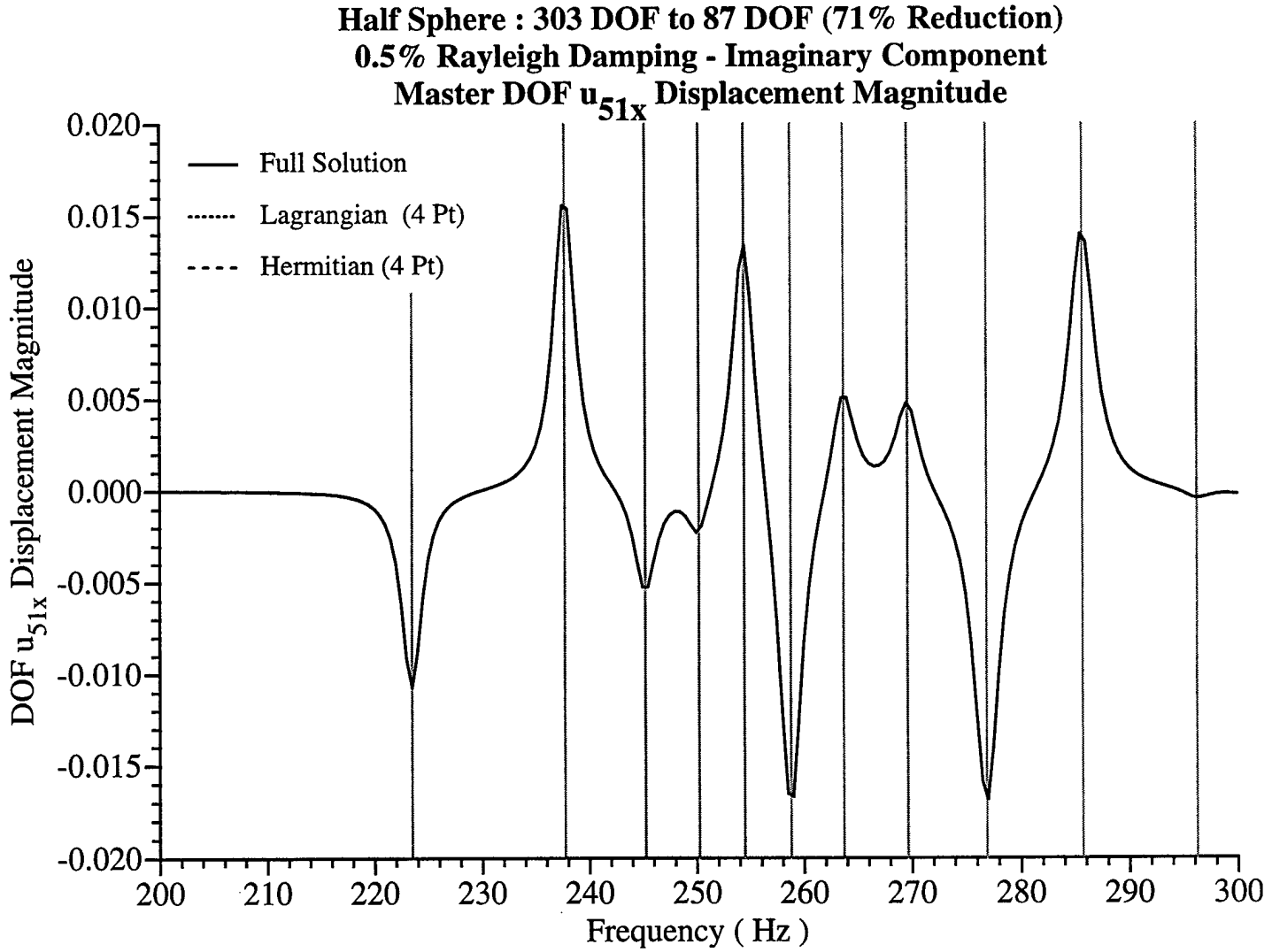


Figure 8.3.53 – The imaginary component of the "master" DOF $|u|_{51x}$. NOTE: 0.5% Damping.

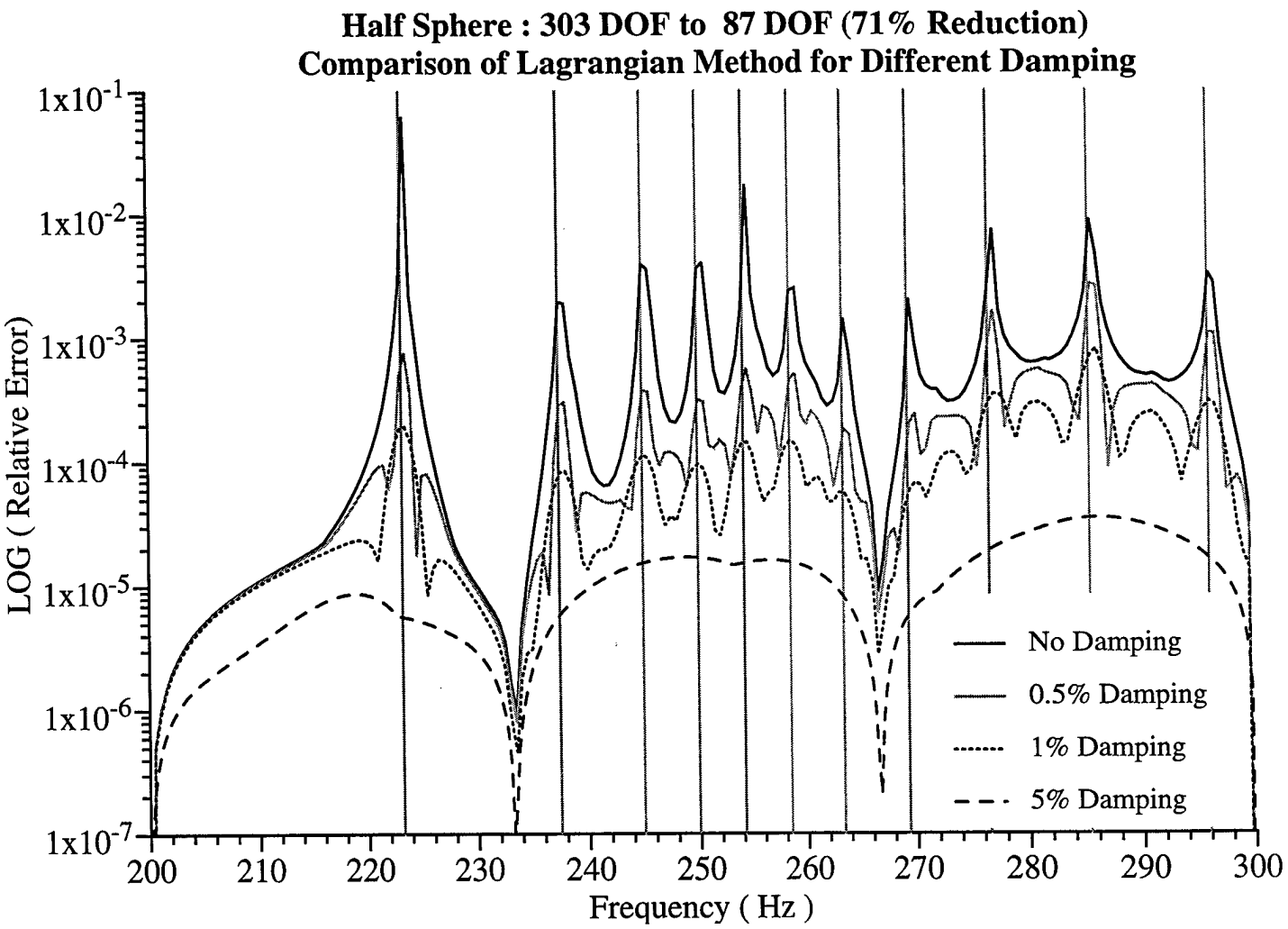


Figure 8.3.54 – Relative Error, real component, comparing level of critical damping for Lagrangian interpolation method for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz.

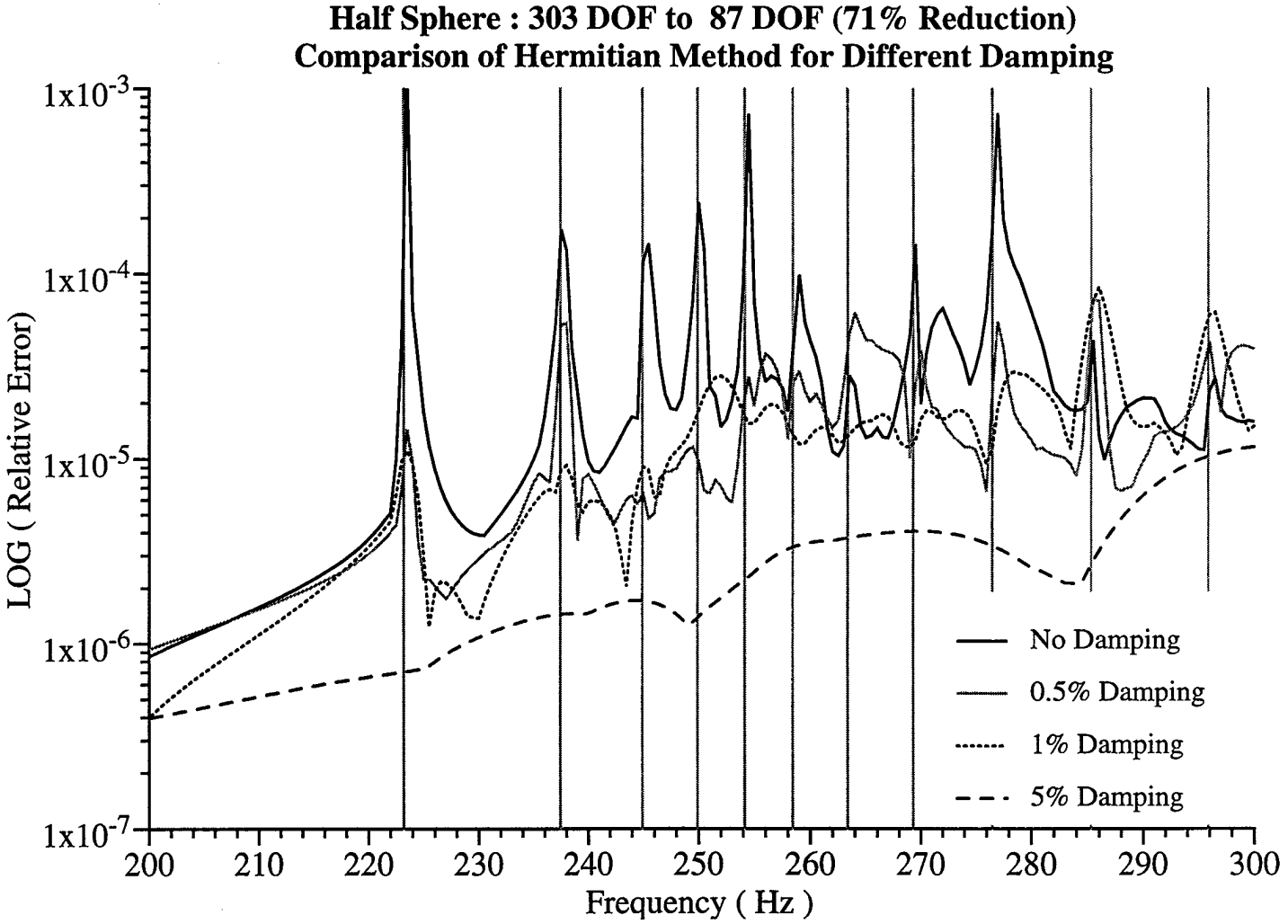


Figure 8.3.55 – Relative Error, real component, comparing level of critical damping for Hermitian interpolation method for the frequency range of 200 to 300 Hz and a frequency step size of 0.5 Hz.

9 Conclusions

The Frequency Window Reduction methodology has been implemented into a finite element environment. The prototype algorithm, developed in F90, has been tested using simple FEM models generated by the ABAQUS™ FEM code. The validity and accuracy of the method has been demonstrated, with excellent results as compare to full DOF model solutions even in the presence of numerous eigenvalues. A number of the analytic features of the FWR methodology have been demonstrated and indicate the utility of these features as analysis tools to extend and enhance frequency analysis as applied to structural acoustics problems. Aspects of the FWR methodology not only have the potential to improve frequency analyses but also to greatly reduce the costs associated with these types of analyses.

The FWR codes having been developed in standard F90 can easily be ported to other computational platforms and may also be extended into a parallel-computing environment. The most general formulation of the FWR method includes complex frequencies, complex frequency interpolation windows and damping. The matrices (mass and stiffness) which may be obtained from third party FEM codes such as ABAQUS allow existing FEM codes to generate the models directly for the FWR method.

Re-ordering according to a user-defined input "master" node list easily allows variable degrees of reduction of models. This offers a means by which the optimum reduction for a given model can be determined and preliminary analyses to be performed in order to characterize the problem. The frequency interpolation window location and size can be selected as well as the variable placement of internal interpolation points within the selected window in order to investigate and then narrow regions of particular interest for the resultant system response. These along with full frequency window sweeps across the interpolation window or partial frequency window sweeps within interpolation window can further refine the analysis. The choice of interpolation methods can be used to examine the system response at appropriate levels of accuracy. The Lagrangian interpolation method is generally less computational work with a slightly reduced accuracy. While the Hermitian interpolation method is more computational work it shows

highly accurate results. The eigenvalues of the full DOF system can also be approximated from the reduced DOF system response, thus allowing the analyses to be done without necessarily requiring an eigenvalue analysis of the full DOF model. The combination of multiple interpolation windows into a larger analysis window shows that the system response of individual smaller windows can be used to build a larger frequency range response tailored to the appropriate levels of accuracy from smaller windows.

The re-use of calculated matrices is an important aspect of the FWR method, which allows multiple analyses to be performed without having to recalculate the necessary matrices each time. For example, given un-ordered mass, and stiffness (and damping) matrices, e.g. from a third party FEM code, it is possible to examine different levels of DOF reduction by changing the input "master" node list. Given this re-ordered set of matrices, a selected interpolation window is used to calculate the constructor matrices, which are then valid for *any* frequency range within the defined interpolation window. That is the constructor matrices do not need to be recalculated as long as the frequency sweep is within the window. Multiple frequency sweep analyses at different resolutions can then easily be performed. Further for a set of constructor matrices defined for an interpolation window, the reduced DOF system matrices may be used to perform analyses for multiple loading and boundary conditions without having to recalculate the constructor matrices. This would allow for a given model variable load analyses to be done quickly without involving a major effort in recalculation.

The varied aspects of the FWR method clearly have the potential to significantly impact the way frequency analyses are performed on large complex models as applied to structural acoustics. The development and implementation of this method as computational tools now offer a more effective and efficient computational environment in which frequency analyses can be performed. The application of these tools should produce a major reduction of computational costs while offering enhanced computational frequency analyses of large complex DOF models.

10 Future Work

The Frequency Window Reduction, FWR, prototype algorithm will continue to be developed and refined as a computational tool. The testing of larger, more complex 2D and 3D structural acoustic models will be undertaken. This will require the FWR codes to be ported to a UNIX environment. With the implementation of the FWR codes into a UNIX environment, it is hoped that other research and/or industrial organizations will assist in the further testing and development of this methodology.

The further investigation of the interpolation functional behavior is needed. The understanding of the interdependencies of the degree of model reduction, the interpolation window location and size, the frequency sweep step size and the choice of the "master" node points is necessary in order to refine the FWR method and its overall efficiency and accuracy. The usage and utility of the "slave-master" interpolation matrix, α, as well as more arbitrary loading and constraint conditions will be investigated as applied to larger FEM models. The matrix properties of the constructor and β matrices need to be characterized in order to develop and implement more efficient and optimize storage and computational schemes. Techniques to obtain the β matrix directly and implement the interpolation of this matrix by similar methods as for G^{-1}will be explored. This would allow more arbitrary loading and constraint conditions on the "slave" DOFs and also lead to reconstruction of the full L^{-1} system operator from the reduced system operator L_0^{-1}.

The utility and implementation of this methodology will be investigated for other frequency dependent applications such as electromagnetic FEM modeling and inverse scattering problems. Finally, the direct incorporation of the resonance or modal response into the computational methodology will be undertaken and tested on FEM models.

11 References

[1] Cook R. D., Malkus D. S., Plesha M. E., Concepts and Applications of Finite Element Analysis, 3rd ed. Wiley, New York (1989).

[2] Seshu P., "Review: Sub-structuring and Component Mode Synthesis", Shock and Vibration 4 (3) 199-210 (1997).

[3] Flippen L. D., "A Theory of Condensation Model Reduction", Computers Math. Applic., 27 9-40 (1994).

[4] Flippen L. D., "Current Dynamics Sub-structuring Methods as Approximations to Condensation Model Reduction", Computers Math. Applic., 27 17-29 (1994).

[5] Dyka C. T., Ingel R. P., Flippen L. D., "A New Approach to Dynamic Condensation for FEM", Computers and Structures, 61 (4) 763-773 (1996).

[6] Flippen L. D., "Interpolation-Based Condensation of Algebraic Semi-Discrete Models with Frequency Response Applications", Computers Math. Applic., 29 (9) 39-52 (1995).

[7] Flippen L. D., "A Generic Bi-Level Formalism for Unifying and Extending Model Reduction Methods", U.S. Naval Research Laboratory Memo Report, submitted (1999).

[8] Bathe K. J., Finite Element Procedures, Prentice-Hall, Englewood Cliffs (1996).

[9] Ellis T. M. R., Philips I. R., Lahey T. M., Fortran 90 Programming, Addison-Wesley, New York (1994).

[10] ABAQUS™ User's, Examples, and Theory Manuals, version 5.6. Hibbitt, Karlsson and Sorensen, Inc., Pawtucket.

[11] Ingel R. P., Dyka C. T., Flippen L. D., "Interpolation-Based Condensation Model Reduction Part 2: Modal Frequency Window Method", ", U.S. Naval Research Laboratory Memo Report, in preparation (1999).

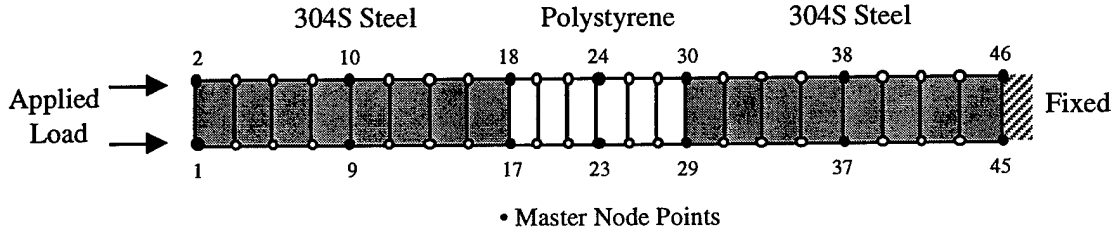
[12] LAPACK Users' Guide, Second Edition, Society for Industrial and Applied Mathematics, Philadelphia, PA. (1995).

12 Appendices

Appendix A - Example of Master and Slave Degrees of Freedom

As described above, the "master" DOFs are chosen as DOFs which have applied loads or fixed boundary conditions. In addition to loaded and constrained DOFs, the master DOFs may be chosen as important place-holders represented in the models; the interfaces or boundaries between element groups of differing material properties or element grouping midpoints. The "slave" degree of freedom refers to the dependent DOFs or those DOFs, which are to be condensed or reduced out of the full FEM model. These are DOFs which typically do not have applied loads or constraints and do not contribute to the system response or are not of interest.

A simple DOF example base on an FEM model is illustrated in the figure below



In this FEM model it is assumed that of the 46 node points there are 14 "master" node points and so there would be 32 "slave" node points. Here the "master" node points are selected as: node points which have the applied load, the fixed boundary condition , the boundaries between element groups of differing material properties and the element grouping midpoints. The original FEM model nodal numbering can then be rewritten via a reordering look-up table as listed below. This table now allows conversion between the original node numbering as is typical in FEM models and the "master" node numbering used in the CMR method.

Node Numbering Look-Up Table

I	II	III
Node Number	"Master" Node Numbering In Original Index	Original Node Numbering In "Master" Index
1	1	1
2	2	2
3	45	15
4	46	16
5	9	17
6	10	18
7	17	19
8	18	20
9	23	5
10	24	6
11	29	21
12	30	22
13	37	23
14	38	24
15	3	25
16	4	26
17	5	7
18	6	8
19	7	27
20	8	28
21	11	29
22	12	30
23	13	9
24	14	10
25	15	31
26	16	32
27	19	33
28	20	34
29	21	11
30	22	12
31	25	35
32	26	36
33	27	37
34	28	38
35	31	39
36	32	40
37	33	13
38	34	14
39	35	41
40	36	42
41	39	43
42	40	44
43	41	45
44	42	46
45	43	3
46	44	4

For the above example the "master" node numbering is input as a list of 14 node numbers from the FEM model which represent the selected nodes which are to be the "master" nodes. (Shaded area in column II of the table) The remainder of the list (Column II, #s 15 to 46) which is automatically generated represents the "slave" nodes, which will be condensed out of the model. The "master" or reordered node numbering correspondence to original or FEM model node numbering which is also automatically generated is listed in column III. The interpretation of this column stated in words is the following... for example node number 3 in the original FEM model (column I) is found via column III to be positioned as node number 15 in the reordered numbering (column II). This look-up table can now be used to either directly assembles the global matrices in the reordered numbering scheme or to reorder existing global matrices.

Appendix B - Frequency Coordinate Transformation

2D Case

If the frequency, ω , is assumed to be a complex variable, $\omega = \omega_1 + i\omega_2$ where the subscript "1" refers to the "real" part of ω and "2" refers to the "imaginary" part of ω, then q is defined as a complex variable, $q = q_1 + iq_2$. It is then possible to define the variable transformation $q \equiv i\omega$ and such that $q = -\omega_2 + i\omega_1$. Hence, $q_1 = -\omega_2$, i.e. q_1 is the "real" part of q, and $q_2 = \omega_1$, i.e. q_2 is the "imaginary" part of q .

A rectangular window in ω-space (frequency window) can be defined by the "real" and "imaginary" origin ω_1 , $i\omega_2$ and the window width, w_ω as defined along the "real" axis and height, h_ω as defined along the "imaginary" axis. Hence, the four vertices of the rectangular window, $\overline{\omega}_i$, are

$$\begin{aligned}
\overline{\omega}_1 &= \omega_1 + i\omega_2 \\
\overline{\omega}_2 &= (\omega_1 + w_\omega) + i\omega_2 \\
\overline{\omega}_3 &= (\omega_1 + w_\omega) + i(\omega_2 + h_\omega) \\
\overline{\omega}_4 &= \omega_1 + i(\omega_2 + h_\omega)
\end{aligned}$$

Note: ω_1, ω_2 and the window height h_ω, and width w_ω, are all real values in ω-space .

Via the coordinate transformation $q \equiv i\omega$, a frequency window may now be referred to in q-space, with an origin q_1, q_2 and a window height, h_q, and width, w_q. The window coordinate origin is then $q = i(\omega_1 + i\omega_2) = -\omega_2 + i\omega_1$, and so the window vertices are

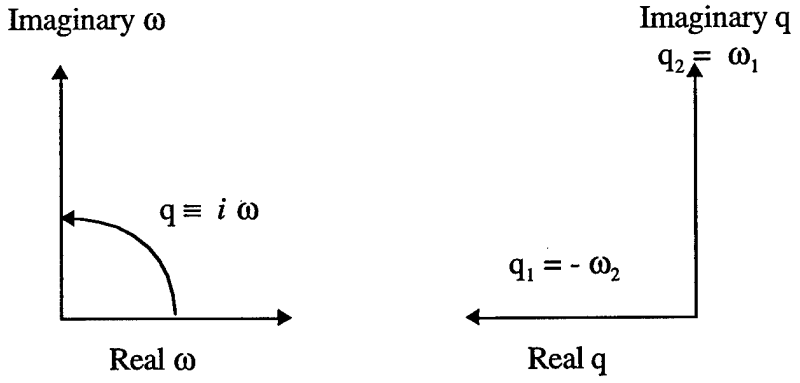
$$\begin{aligned}
\overline{q}_1 &= i\overline{\omega}_1 = -\omega_2 + i\omega_1 \\
\overline{q}_2 &= i\overline{\omega}_4 = -(\omega_2 + h_\omega) + i\omega_1 \\
\overline{q}_3 &= i\overline{\omega}_3 = -(\omega_2 + h_\omega) + i(\omega_1 + w_\omega) \\
\overline{q}_4 &= i\overline{\omega}_2 = -\omega_2 + i(\omega_1 + w_\omega)
\end{aligned}$$

Note: q_1, q_2 and the window height, h_q, and width, w_q, are all real values in q-space.

Given that $q_1 = -\omega_2$ and $q_2 = \omega_1$ and also $w_q = h_\omega$ and $h_q = w_\omega$ the window vertices are

$$\begin{aligned}
\overline{q}_1 &= q_1 + i\, q_2 \\
\overline{q}_2 &= (q_1 + w_q) + i\, q_2 \\
\overline{q}_3 &= (q_1 + w_q) + i\,(q_2 + h_q) \\
\overline{q}_4 &= q_1 + i\,(q_2 + h_q)
\end{aligned}$$

The coordinate transformation $q \equiv i\omega$, may be simply thought of as a rotation of the frequency axis by 90°.



1D Case

For the 1D case, the interpolation points of a frequency range or line, $\overline{\omega}_j(\omega_1, \omega_2, w_\omega, h_\omega) = \omega_0, \omega_e$ are defined as the line end points

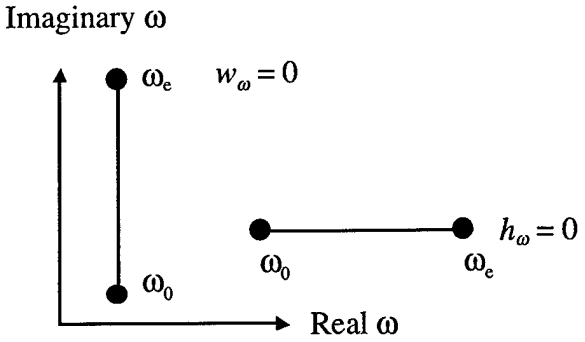
$$\bar{\omega}_0 = \omega_1 + i\omega_2$$

either

$$\bar{\omega}_e = (\omega_1 + w_\omega) + i\omega_2 \qquad \text{if } h_\omega = 0$$

or

$$\bar{\omega}_e = \omega_1 + i(\omega_2 + h_\omega) \qquad \text{if } w_\omega = 0$$



As above for the 2D case via the coordinate transformation $q \equiv i\omega$, a frequency line may now be referred to in q-space, with an origin q_0, q_e and a window height, h_q, and width, w_q. The line coordinate end points in q-space are then found as

$$\bar{q}_0 = i\bar{\omega}_0 = -\omega_2 + i\,\omega_1$$
$$\bar{q}_e = i\bar{\omega}_e = -(\omega_2 + h_\omega) + i\,\omega_1 \qquad \text{if } w_\omega = 0$$
$$\bar{q}_e = i\bar{\omega}_w = -\omega_2 + i\,(\omega_1 + w_\omega) \qquad \text{if } h_\omega = 0$$

Appendix C – Reduced Mass, Stiffness and Damping Matrices

A frequency $\omega = \omega_1 + i\omega_2$ is transformed to the generalize variable $q = q_1 + iq_2$ by $q \equiv i\omega$. Then for a given value of $\hat{q} = f(\hat{q}_1, \hat{q}_2)$ within the interpolation window, the $\beta(\hat{q})$ matrix may be founds as

$$\beta(\hat{q}) \cong \hat{G}^{-1}(\hat{q})\; H(\hat{q})$$

Where $H(\hat{q}) = \alpha(\hat{q})\left(\hat{q}^2 M_{11} + \hat{q} D_{11} + S_{11}\right) - \left(\hat{q}^2 M_{21} + \hat{q} D_{21} + S_{21}\right)$.

The $\hat{G}^{-1}(\hat{q})$ matrix for Hermitian Interpolation for a given value of $\hat{q} = f(\hat{q}_1, \hat{q}_2)$ within the window having previously obtained the $W(\overline{q}_j)$ for the rectangular window at the points $\overline{q}_j$ is

$$\hat{G}^{-1}(\hat{q}) \cong C_0(\hat{q}_1, \hat{q}_2)\ W(\overline{q}_j)$$

Alternatively the value of $\hat{G}^{-1}(\hat{q})$ can be interpolated by the Shape Functions evaluated at $\hat{q} = f(\hat{q}_1, \hat{q}_2)$ from the isoparametric coordinates $(\hat{r}, \hat{s})$ and the $G^{-1}(\overline{q}_j)$ matrices previously calculated at each of the "element" or window interpolation points by

$$\hat{G}^{-1}(\hat{q}) = \sum_{i=1}^{9} h_i(\hat{r}, \hat{s})\ G_i^{-1}(\overline{q})$$

Finally reduced system operator, $\overline{\mathbf{L}}_0(q)$ is

$$\begin{aligned}\tilde{\mathbf{L}}_0(\hat{q}) &= \tilde{\mathbf{L}}_{11}(\hat{q}) + \tilde{\mathbf{L}}_{12}(\hat{q})\ \beta(\hat{q}) \\ &= \left(\hat{q}^2\mathbf{M}_{11} + \hat{q}\mathbf{D}_{11} + \mathbf{S}_{11}\right) + \left(\hat{q}^2\mathbf{M}_{12} + \hat{q}\mathbf{D}_{12} + \mathbf{S}_{12}\right)\ \beta(\hat{q})\end{aligned}$$

From this equation, the reduced mass, stiffness and damping matrices may be obtained as

$$\begin{aligned}\overline{\mathbf{M}}_0(\hat{q}) &= \mathbf{M}_{11} + \mathbf{M}_{12}\ \beta(\hat{q}) \\ \overline{\mathbf{S}}_0(\hat{q}) &= \mathbf{S}_{11} + \mathbf{S}_{12}\ \beta(\hat{q}) \\ \overline{\mathbf{D}}_0(\hat{q}) &= \mathbf{D}_{11} + \mathbf{D}_{12}\ \beta(\hat{q})\end{aligned}$$

Hence in order to calculate the reduced mass, stiffness and damping matrices the matrix $\beta(\hat{q})$must be obtained at each specific value of the frequency (or $\hat{q}$).